

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Факультет інформатики та обчислювальної техніки

(повне найменування інституту, факультету)

Автоматизованих систем обробки інформації і управління

(повна назва кафедри)

«До захисту допущено»

В.о. завідувача кафедри

(підпис) О.А.Павлов
(ініціали, прізвище)

“ ” 2019 р.

Дипломний проект

на здобуття ступеня бакалавра

з напрямку підготовки 6.050103 «Програмна інженерія»
на тему Сервіс аналізу ринку іноземних валют

Виконав: студент IV курсу, групи ІП-52 Ясенова Анна Вадимівна
(прізвище, ім'я, по батькові) _____ (підпис)

Керівник ст.в. кафедри АСОІУ Халус О.А.
посада,науковий ступінь,вчене звання,прізвище,ініціали _____ (підпис)

**Консультант
з графічної
документації** доц., к.т.н., Ліщук К.І.
посада,науковий ступінь,вчене звання,прізвище,ініціали _____ (підпис)

Рецензент:

посада,науковий ступінь,вчене звання,прізвище,ініціали _____ (підпис)

Засвідчую, що у цьому дипломному проекті
немає запозичень з праць інших авторів без
відповідних посилань.

Студент _____
(підпис)

Київ – 2019 року

**Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”**

Факультет (інститут) Інформатики та обчислювальної техніки
(повна назва)

Кафедра автоматизованих систем обробки інформації і управління
(повна назва)

Рівень вищої освіти – перший (бакалаврський)

Напрямок підготовки (програма професійного спрямування) – 6.050103
«Програмна інженерія» (Програмне забезпечення систем)

ЗАТВЕРДЖУЮ

В.о. завідувача кафедри

О.А. Павлов
(підпис) (ініціали, прізвище)

“ ” 2019 р.

**ЗАВДАННЯ
НА ДИПЛОМНИЙ ПРОЕКТ СТУДЕНТУ**

Ясеновій Анні Вадимівній

(прізвище, ім'я, по батькові)

1. Тема проекту «Сервісу аналізу ринку іноземних валют»

керівник проекту Халус О.А., ст. викладач
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від “23” квітня 2019 р. №1181-с

2. Термін подання студентом проекту «03» червня 2019 року

3. Вихідні дані до проекту

Технічне завдання

4. Зміст пояснювальної записки

1. Аналіз вимог до програмного забезпечення: опис предметного середовища, огляд існуючих технічних рішень та відомих програмних продуктів, розробка

функціональних та нефункціональних вимог, математичне забезпечення

2. Моделювання та конструювання програмного забезпечення: моделювання та аналіз програмного забезпечення, архітектура програмного забезпечення, конструювання програмного забезпечення, аналіз безпеки даних

3. Аналіз якості та тестування програмного забезпечення: аналіз якості ПЗ, опис процесів тестування, опис контрольного прикладу

4. Впровадження та супровід програмного забезпечення: розгортання програмного забезпечення, робота з програмним забезпеченням

5. Перелік графічного матеріалу

1. Схема структурна варіантів використання

2. Схема структурна класів програмного забезпечення

3. Креслення вигляду екранних форм

6. Консультанти розділів проекту

| Розділ | Прізвище, ініціали та посада консультанта | Підпис, дата | |
|--------|---|----------------|------------------|
| | | завдання видав | завдання прийняв |
| | | | |

7. Дата видачі завдання «12» березня 2018 року

КАЛЕНДАРНИЙ ПЛАН

| № з/п | Назва етапів виконання дипломного проекту | Термін виконання етапів проекту | Примітка |
|-------|--|---------------------------------|----------|
| 1. | Вивчення рекомендованої літератури | 19.03.2019 | |
| 2. | Аналіз існуючих методів розв'язання задачі | 26.03.2019 | |
| 3. | Постановка та формалізація задачі | 26.03.2019 | |
| 4. | Аналіз вимог до програмного забезпечення | 02.04.2019 | |
| 5. | Алгоритмізація задачі | 02.04.2019 | |
| 6. | Моделювання програмного забезпечення | 09.04.2019 | |
| 7. | Обґрунтування використовуваних технічних засобів | 16.04.2019 | |
| 8. | Розробка архітектури програмного забезпечення | 23.04.2019 | |
| 9. | Розробка програмного забезпечення | 30.04.2019 | |
| 10. | Налагодження програми | 07.05.2019 | |
| 11. | Виконання графічних документів | 14.05.2019 | |
| 12. | Оформлення пояснювальної записки | 21.05.2019 | |
| 13. | Подання ДП на попередній захист | 28.05.2019 | |
| 14. | Подання ДП рецензенту | 03.05.2019 | |
| 15. | Подання ДП на основний захист | 08.06.2019 | |

Студент _____ Ясенова А.В.
(підпис)

Керівник проекту _____ Халус О.А.
(підпис)

[illegible]

АНОТАЦІЯ

Пояснювальна записка дипломного проекту складається з чотирьох розділів, містить 35 таблиці, 3 рисунків, 1 додаток та 7 джерел – загалом 84 сторінки.

Об'єкт дослідження: сервіси надання послуг на ринку іноземних валют.

Мета дипломного проекту: надання аналітичних послуг у сфері ринку іноземних валют, акумуляція в одному сервісі інформацію з різних джерел і проведення статистичного аналізу курсу валют на різних рівнях.

У першому розділі проведено аналіз відомих технічних рішень і розроблені вимоги до програмного забезпечення. Побудовано схему структурну варіантів використання.

У другому розділі було розроблено архітектуру сервісу як комп'ютерного додатку. Побудовано структурну схему класів.

У третьому розділі проведено тестування сервісу аналізу ринку іноземних валют за розробленим планом тестування. Описано процес тестування.

У четвертому розділі описано розгортання та впровадження сервісу, створено керівництво користувача.

У додатках наведено: текст програмного коду.

КЛЮЧОВІ СЛОВА: ФОРЕКС, РИНОК ІНОЗЕМНИХ ВАЛЮТ, ТОРГІВЛЯ, КУРС ВАЛЮТИ

ABSTRACT

The explanatory note of the diploma project consists of four sections, 35 tables, 3 illustrations, 1 appendix and 7 sources – totally 84 pages.

The object of study: applications for providing services on the foreign exchange market.

The aim of the diploma project: to provide analytical services for the foreign exchange market, accumulate information from different sources in one place and conduct statistical analysis of the exchange rate at different levels.

In the first section the analysis of known technical solutions was provided and software requirements were developed. The use-case diagram was constructed.

In the second section the architecture of the service as desktop application was described and developed. A structural diagram of classes is constructed.

In the third section, the application was tested according to the developed test plan. The process of testing was described.

The fourth section describes the deployment and implementation of the application. The user guide was provided.

Appendices contain the description of the program.

KEYWORDS: FOREX, FOREIGN EXCHANGE MARKET, TRADING, CURRENCY RATE

Пояснювальна записка до дипломного проекту

на тему: Сервіс аналізу ринку іноземних валют

Київ – 2019 року

ЗМІСТ

| | |
|---|-----------|
| ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ | 10 |
| ВСТУП..... | 11 |
| 1 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ | 14 |
| 1.1 ЗАГАЛЬНІ ПОЛОЖЕННЯ | 14 |
| 1.2 ЗМІСТОВНИЙ ОПИС І АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ | 14 |
| 1.3 АНАЛІЗ УСПІШНИХ ІТ-ПРОЕКТІВ..... | 15 |
| 1.3.1 Аналіз відомих технічних рішень | 15 |
| 1.3.2 Аналіз відомих програмних продуктів..... | 17 |
| 1.4 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ | 19 |
| 1.4.1 Розроблення функціональних вимог..... | 19 |
| 1.4.2 Розроблення нефункціональних вимог | 30 |
| 1.4.3 Постановка комплексу завдань модулю | 31 |
| 1.5 МАТЕМАТИЧНЕ ЗАБЕЗПЕЧЕННЯ..... | 32 |
| 1.6 Висновки до розділу | 35 |
| 2 МОДЕЛЮВАННЯ ТА КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ..... | 36 |
| 2.1 МОДЕЛЮВАННЯ ТА АНАЛІЗ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ..... | 36 |
| 2.2 АРХІТЕКТУРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ | 37 |
| 2.3 КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ..... | 54 |
| 2.4 АНАЛІЗ БЕЗПЕКИ ДАНИХ..... | 56 |
| 2.5 Висновки до розділу | 57 |
| 3 АНАЛІЗ ЯКОСТІ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ | 58 |
| 3.1 АНАЛІЗ ЯКОСТІ ПЗ..... | 58 |
| 3.2 ОПИС ПРОЦЕСІВ ТЕСТУВАННЯ..... | 59 |
| 3.3 ОПИС КОНТРОЛЬНОГО ПРИКЛАДУ | 72 |
| 3.4 Висновки до розділу | 73 |
| 4 ВПРОВАДЖЕННЯ ТА СУПРОВІД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ | 74 |
| 4.1 РОЗГОРТАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ..... | 74 |
| 4.2 РОБОТА З ПРОГРАМНИМ ЗАБЕЗПЕЧЕННЯМ..... | 74 |

| | | |
|---|---------------------------|-----------|
| 4.3 | Висновки до розділу | 75 |
| ВИСНОВКИ | | 76 |
| ПЕРЕЛІК ПОСИЛАНЬ..... | | 77 |
| ДОДАТОК А ТЕКСТ ПРОГРАМНОГО КОДУ | | 78 |

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 9 |

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

Форекс (ринок іноземних валют) – це позабіржевий міжбанківський ринок, що являє собою міжнародну неформальну мережу торгових відносин, учасниками яких є як приватні особи, так і банки, та іншу фінансові установи.

Трейдер – це особа, яка торгує активами на біржі з метою отримання прибутку.

Диверсифікація (новолат. diversificatio — зміна, різноманітність; від лат. diversus — різний і facere — робити) — володіння найрізноманітнішими фінансовими активами, кожен з яких має різний рівень ризику, з метою зниження загального ступеня ризику портфеля в цілому [1].

Тренд в економіці - напрямок переважного руху показників. Зазвичай розглядається в рамках технічного аналізу, де мають на увазі спрямованість руху цін або значень індексів [2].

Ліва нога валютної пари – це базисна валюта, та, в якій визначається курс валютної пари і відносно якої котирується друга валюта. В назві валютної пари записується зліва.

Права нога валютної пари – це валюта котирування. В назві валютної пари записується справа.

Волатильність – це показник, який характеризує тенденцію ринкової ціни або доходу змінюватися з часом і є мірою ризику використання фінансового інструмента за заданий проміжок часу [3].

Коефіцієнт Шарпа (Sharp Ratio) – це показник ефективності інвестиційного активу, який обчислюється як відношення середньої премії за ризик до стандартного відхилення вартості активу [4].

Просадка (Drawdown) - це відхилення балансу/вартості торгового акаунта/активу відносно максимального історичного (пікового) значення [5].

ВСТУП

Ринок іноземних валют, або форекс, з'явився дуже давно, але масова торгівля на ньому стала популярною відносно недавно. В зв'язку з тим, що трейдери та інвестори прагнуть диверсифікувати свої ризики виникають і розвиваються різні методи аналізу курсів тих чи інших валют, індексів та інших активів: статистичні оцінки, технічний аналіз, нейронні мережі та багато інших.

Сьогодні ми спостерігаємо золоту добу в розвитку ІТ сфери: досить важко знайти сервіс, магазин чи сферу, яка не має свого веб-сервісу чи іншого програмного продукту. Відповідно, наприкінці 20 сторіччя з'являються перші програмні продукти для ринку іноземних валют. Серед них спочатку були ті, які забезпечують внутрішні операції і не призначені для трейдерів. Пізніше з'явилися продукти, призначенням яких було надання можливості торгувати на ринку. І лише протягом останніх 10-15 років, в час розквіту таких напрямків як аналіз даних, Data Science та Machine Learning з'являються додатки і сервіси, основним призначенням яких є надання аналітичних послуг, швидкого доступу до інформації, агрегація даних, аналіз курсу валют і вартості активів, оцінка торгівлі трейдерів.

На жаль, більшість таких сервісів пропонують обмежений функціонал (лише конвертація, або лише завантаження інформації), а якщо і надають можливість отримати аналітичну оцінку, то лише для курсу валюти з одного джерела даних. Чому це є проблемою? Справа в тому, що різні джерела курсу валют надають різну інформацію. Наприклад, банки матимуть відмінні показники для одних і тих же валютних пар, якщо вони розміщені в різних країнах, так як на коливання вартості активу впливають і відсоткова ставка банку, і зовнішні умови (події в країні, стабільність економіки і т.д.).

У даному дипломному проекті розроблений сервіс аналізу ринку іноземних валют, призначений для швидкого отримання інформації про курс валютної пари та її аналізу з одного додатку для різних джерел інформації, та, як наслідок зменшення часових витрат на процес торгівлі і надання можливості

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| | | | | | | 11 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

для порівняння цін на активи як всередині одного джерела інформації (банк, брокер), так і між ними.

Розроблений сервіс може застосовуватися трейдерами які торгують на одній чи кількох платформах будь-якою кількістю валютних пар, так як для отримання інформації не потрібно перемикатися між різними сервісами, варто лише змінити один чи два параметри і оновити результат.

Актуальність теми: стрімке зростання популярності торгівлі на ринку іноземних валют і відсутність сервісів, що агрегують інформацію з різних джерел.

Мета розробки: надання аналітичних послуг у сфері ринку іноземних валют, акумуляція в одному сервісі інформації з різних джерел і проведення статистичного аналізу курсу валют на різних рівнях, спрощення процесу отримання та аналізу даних про конкретний стан ринку іноземних валют і зменшення витрат часу на пошук необхідної інформації.

Завдання розробки: ведення даних з різних сервісів, обчислення аналітичних показників, налаштування візуалізації, отримання даних, ведення джерел інформації, статистичний аналіз даних за обраний період, обчислення вартості конвертації між валютами .

Практичне значення одержаних результатів: розроблений сервіс реалізований у вигляді десктопного додатку і надає можливість використовуючи лише один програмний продукт отримувати інформацію про курс валюти, статистичну оцінку та візуалізацію статистик використовуючи інформацію з різних джерел.

Публікації:

– Ясенова А.В. Диверсифікація ризиків на ринку іноземних валют // II Всеукраїнська науково-практична конференція молодих вчених та студентів «Інформаційні системи та технології управління – ІСТУ-2019» ;

– Ясенова А.В. Дослідження просадок курсу іноземних валют // Міжнародна науково-практична конференція «Економіка, фінанси, облік та право: стратегічні пріоритети розвитку в умовах глобалізації».

1 АНАЛІЗ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

1.1 Загальні положення

У життєвому циклі сучасних технологій чи не найважливішим етапом є збір та аналіз вимог, протягом якого необхідно зафіксувати вимоги до системи та її розробки, визначити функції, терміни та вартість робіт та показників якості, яких необхідно досягнути в процесі розробки. Ці вимоги вважаються розробленими коректно, якщо вони відповідають потребам осіб, що зацікавлених у створенні програмного забезпечення — клієнти, користувачі, замовники.

Успішність торгівлі трейдера на ринку іноземних валют значною мірою залежить від програмного забезпечення, яке він використовує. Важливо, щоб таке програмне забезпечення було актуальним, тобто орієнтованим на реалії сучасного світу. Тому необхідно провести збір та аналіз вимог до такого програмного забезпечення і узгоджувати увесь подальший процес розробки з цими вимогами.

1.2 Змістовний опис і аналіз предметної області

Сьогодні робота на фінансовому ринку - це війна технологій. Для того, щоб полегшити процес торгівлі створено велику кількість програмних продуктів, серед яких можна виділити кілька основних типів за призначенням:

- торгові платформи - програмне забезпечення, яке надає доступ до актуального курсу валют і включає в себе найпростіші та відображення графіків (зазвичай це коливання курсу і торгові сигнали). Власне, таке програмне забезпечення являє собою мінімальний необхідний набір інструментів для здійснення торгової транзакції (купівля чи продаж). Основною вимогою до цих програм є актуальність даних, тобто відсутність затримок;

- порадики форекс - це автоматичні торгові системи - програми, які торгують самостійно без сторонньої допомоги і керуються лише власним

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 14 |

алгоритмом. Основними вимогами до них є коректність роботи і здатність приносити прибуток;

— програми для торгових сигналів - програмне забезпечення, основною метою якого є донести торговий сигнал до трейдера чи будь-якого іншого учасника торгівлі. Сигнали можуть надсилатися від брокерів до клієнтів або генеруватися характерними змінами в курсі валютної пари (чи іншого торгового активу). Основною вимогою до такого програмного забезпечення є актуальність і коректність сигналів, адже якщо повідомлення надіслане клієнту є застарілим чи невірним, то, власне, обличчя продукту втрачається. Менш критичною, але також дуже важливою для таких додатків є взаємодія з користувачем: повідомлення повинні надсилатися з правильною частотою (не надто часто, адже тоді програма набридне клієнту і він банально вимкне її, і не надто рідко, бо в такому випадку клієнт не відреагує вчасно на сигнал і всі прагнення отримати користь від програми зводяться на нівець);

— індивідуальні форекс роботи - програми, які в теорії не відрізняються від описаних вище, проте на практиці мають основну відмінність: вони не є у вільному доступі, тобто це той же поради́ник, але працює він вже за власним алгоритмом трейдера, а отже трейдер має створити цей поради́ник власноруч, або заплатити розробникам за його створення. Разом з цим, у трейдерів з'являється можливість торгувати використовуючи власні алгоритми.

1.3 Аналіз успішних ІТ-проектів

1.3.1 Аналіз відомих технічних рішень

Однією з найбільш важливих вимог до програмного забезпечення, що використовується для торгівлі на ринку іноземних валют є актуальність даних, тому гостро стоїть питання про регулярність їх оновлення. На сьогоднішній день майже всі онлайн-сервіси надають доступ до своїх даних через API-інтерфейс чи веб-сокети. Тому стало досить поширеною практикою написання власних

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КП.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 15 |

обгортку до наданих інтерфейсів на обраній мові програмування, щоб надалі використовувати її для взаємодії зі сторонніми сервісами. Для регулярного оновлення даних в базі даних сервісу добре працює будь-яка системна утиліта, яка дозволяє автоматично виконувати команди в заданий час (наприклад cron для Linux і Unix операційних систем).

Аналіз даних на фореکсі не потребує використання принципово відмінних від класичних інструментів автоматизації. Навпаки, з часом з'являється все більше вдосконалень, бібліотек і фреймворків, які полегшують цей процес. Стандартними інструментами для аналізу даних є MATLAB, R і Python.

MATLAB являє собою набір програм і мову програмування для вирішення чисельних та технічних задач і чудово підходить для розробки стратегій та їх бектесту (backtest - тестування стратегії чи будь-якого іншого алгоритму на даних з минулого).

R - це мова програмування для статистичного аналізу даних і їх візуалізації, яка ідеально підходить для ситуацій, коли результати потрібно отримати швидко ("на колінці"), адже це скриптова мова, яка оптимізована для виконання математичних операцій різної складності тому працює швидко, а також якщо потрібно провести розширений аналіз або підготувати більш складні візуалізації. R доступна для використання в інших мовах програмування (наприклад в Python за допомогою пакету RPy), і навпаки, програмний код, написаний на інших мовах можна запустити з R, тому отримані результати можна легко використовувати в подальшій розробці. Одним з найкращих і найбільш повних інструментів для математичних обчислень для ринку іноземних валют в мові R є пакет PerformanceAnalytics, що включає в себе економетричні функції для аналізу ефективності та ризикованості фінансових інструментів і портфелів.

Python - інтерпретована мова програмування, яка має неймовірно велику кількість потужних засобів і пакетів для аналізу даних та математичних обчислень:

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КП.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 16 |

- numpy для роботи з багатовимірними масивами та реалізації і використання математичних функцій різної складності;
- scipy для розв’язання таких задач оптимізації, диференціальних рівнянь, а також для обробки зображень і реалізації генетичних алгоритмів;
- pandas для різного роду маніпуляцій з даними (у вигляді таблиць або часових рядів) та їх аналізу;
- matplotlib, seaborn, plotly і bokeh для візуалізації даних від найпростіших лінійних до багатовимірних залежностей;
- scikit-learn, tensorflow і keras для побудови складних регресійних моделей, розв’язання задач оптимізації або машинного навчання.

Також для Python розроблена інтерактивна оболонка IPython, яка дуже активно використовується для наукової розробки та досліджень в таких сферах як Data Science та аналіз даних. Вона надає велику кількість засобів для візуалізації даних та маніпуляцій з ними і поєднує в собі можливості командного рядка Unix та інтерактивного командного рядка Python.

Для виконання даного дипломного проекту використана мова програмування Python для написання обгортки до API-інтерфейсів джерел даних, а також такі бібліотеки як pandas, scipy, numpy для обчислень і аналізу даних, matplotlib для візуалізації даних.

1.3.2 Аналіз відомих програмних продуктів

Найбільш поширена на сьогодні торгова платформа - це MetaTrader. Вона дозволяє завантажувати власні дані для аналізу або підключатися до сторонніх сервісів для отримання даних в реальному часі, дає доступ до багатьох необхідних для торгівлі графіків і дає можливість торгувати активами і валютними парами, створювати демо-рахунки для тестування і перевірки алгоритмів, порадників, припущень. Також розроблена C-подібна мова програмування MQL, яка дозволяє створювати порадників, торгових роботів та інші інструменти для автоматизації процесу торгівлі всередині платформи.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КП.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 17 |

Разом з цим, критики відмічають, що даний продукт має технічні можливості, які дозволяють брокерам вводити клієнтів в оману або виконувати не вигідні їм торгові операції. Також вона не працює для сімейства Unix операційних систем, потребує відкриття хоча б демо-акаунта для того, щоб мати доступ до будь-якої інформації і не надає аналітичної оцінки курсу валютної пари.

Однією з перших платформ для аналізу поведінки і оцінки успішності трейдерів є myfxbook.com, яка, в першу чергу, призначена для автоматичного аналізу торгових рахунків і формування спільноти трейдерів. На даний момент сервіс підтримує понад 100 різних брокерів, дає можливість аналізувати, порівнювати і поширювати результати своєї торгової діяльності. Сайт має дуже широкий спектр функцій і можливостей, таких як пошук виконавців для замовлень торгових порадників і роботів, публікація нових і перегляд старих візуалізацій та багато інших. Принциповим недоліком сервісу є відверто поганий дизайн, адже він є застарілим і незручним в користуванні, а також надто велика кількість функціоналу, через яку втрачається розуміння основного призначення сайту; надані користувачу можливості варто хоча б групувати за категоріями, що вже значною мірою спростить навігацію сайтом.

Менш популярними є новіші сервіси PowerStats і Autochartist. Проте, вони є більш зручними в користуванні, подають інформацію в дуже зручному вигляді і мають інтуїтивно зрозумілий інтерфейс. Саме ці продукти стануть лідерами серед аналітичних продуктів в сфері торгівлі в найближчі 10 років. Autochartist є одним із найбільш розвинутих засобів для розпізнавання патернів, доступний як браузерний додаток і плагін для MetaTrader, яка пропонує PowerStats як статичний додаток, який надає інформацію про ринок. Для звичайних користувачів Autochartist є платною програмою, безкоштовною вона буде, якщо відкрити рахунок у брокера-партнера, також є можливість скористатися пробним 14-денним періодом.

1.4 Аналіз вимог до програмного забезпечення

Система повинна містити такі типи користувачів як адміністратор і користувач.

Адміністратор – це користувач з особливими правами, має доступ до сторонніх джерел інформації для завантаження даних з них до бази даних системи.

Користувач має доступ до функціоналу сервісу: може експортувати дані, переглядати візуалізації аналітичних показників, розраховувати вартість конвертації різних валют одна між одною.

В системі повинні бути реалізовані наступні функції:

- перегляд візуалізацій аналітичних показників;
- статистичний аналіз курсу валюти;
- експорт інформації про курс валют до файлу;
- розрахунок вартості конвертації між валютами.

1.4.1 Розроблення функціональних вимог

Схема структурна варіантів використання наведена у графічному матеріалі.

В системі передбачено наступні варіанти використання, описані у таблицях 1.1 – 1.27:

Таблиця 1.1 - Варіант використання UC001

| | |
|----------|---|
| Назва | Ведення джерел інформації |
| Опис | Адміністратор може додавати і видаляти (приховувати) джерела інформації вручн |
| Учасники | Адміністратор |

Таблиця 1.2 - Варіант використання UC002

| | |
|-------------------|---|
| Назва | Додавання джерел інформації |
| Опис | Адміністратор може додавати джерело інформації про курс валют вручну |
| Учасники | Адміністратор |
| Передумови | Налагоджена взаємодія з джерелом, що надає дані – створено і протестовано обгортку для API-інтерфейсу, інформація завантажена до бази даних сервісу |
| Постумови | Джерело інформації додано на всіх діючих сторінках додатку |
| Основний сценарій | Адміністратор вручну додає джерело даних до списку тих, що відображаються користувачеві |

Таблиця 1.3 - Варіант використання UC003

| | |
|-------------------|--|
| Назва | Видалення джерел інформації |
| Опис | Адміністратор може видаляти джерело інформації про курс валют вручну |
| Учасники | Адміністратор |
| Передумови | Виникли складнощі в роботі сервісу, що надає інформацію чи взаємодії з ним; зникла необхідність використання даних з сервісу |
| Постумови | Джерело інформації не доступне для вибору користувачам на всіх діючих сторінках додатку |
| Основний сценарій | Адміністратор вручну видаляє джерело даних зі списку тих, що відображаються користувачеві |

Таблиця 1.4 - Варіант використання UC004

| | |
|-------------------|--|
| Назва | Завантаження даних |
| Опис | Адміністратор може завантажувати інформацію про курси валютних пар до бази даних сервісу за допомогою завчасно підготованих скриптів |
| Учасники | Адміністратор |
| Передумови | Налагоджена взаємодія з джерелом, що надає дані – створено і протестовано обгортку для API-інтерфейсу |
| Постумови | Інформація про курси валют завантажена до бази даних сервісу |
| Основний сценарій | Адміністратор запускає скрипт, який надсилає запит до джерела даних, отримує інформацію і записує її до бази даних |

Таблиця 1.5 - Варіант використання UC005

| | |
|-------------------|---|
| Назва | Перегляд результатів статистичного аналізу. |
| Опис | Користувач може переглянути результати статистичного аналізу курсу валютної пар |
| Учасники | Користувач |
| Передумови | З'явилася потреба дізнатися додаткову інформацію про курс валюти в минулому, необхідна для аналізу інформація наявна в базі даних сервісу |
| Постумови | Результати статистичного аналізу виведені на екран. |
| Основний сценарій | Користувач перейшов з головної сторінки на сторінку перегляду статистичних результатів, обрав коректні вхідні дані, натиснув кнопку “Analyze” |

Продовження таблиці 1.5

| | |
|----------------------|---|
| Розширення сценаріїв | Користувач може змінювати одну з валютних пар або період для аналізу і порівнювати результати для різних випадків, вони розмістяться один під одним |
|----------------------|---|

Таблиця 1.6 - Варіант використання UC006

| | |
|-------------------|--|
| Назва | Вибір валютної пари |
| Опис | Користувач може обирати валютну пару (ліву і праву ногу) для перегляду візуалізації чи статистики, розрахунку конвертації чи експорту інформації |
| Учасники | Користувач |
| Передумови | Користувач перейшов на одну з функціональних сторінок |
| Постумови | Обрана одна валютна пара із запропонованих |
| Основний сценарій | Користувач переходить на одну з функціональних сторінок сервісу і обирає ліву і праву ногу валютної пари |

Таблиця 1.7 - Варіант використання UC007

| | |
|------------|--|
| Назва | Вибір джерела інформації |
| Опис | Користувач може обирати джерело інформації для перегляду візуалізації чи статистики, розрахунку конвертації чи експорту інформації |
| Учасники | Користувач |
| Передумови | Користувач перейшов на одну з функціональних сторінок |
| Постумови | Джерело інформації обране |

Продовження таблиці 1.7

| | |
|-------------------|---|
| Основний сценарій | Користувач переходить на одну з функціональних сторінок сервісу і обирає джерело даних для подальших дій над ними |
|-------------------|---|

Таблиця 1.8 - Варіант використання UC008

| | |
|----------------------|---|
| Назва | Вибір періоду часу для аналізу |
| Опис | Користувач може обирати періоду часу для аналізу для перегляду візуалізації чи статистики і експорту інформації |
| Учасники | Користувач |
| Передумови | Користувач перейшов на одну з функціональних сторінок |
| Постумови | Період для аналізу даних встановлений |
| Основний сценарій | Користувач переходить на одну з функціональних сторінок сервісу і обирає період для аналізу даних |
| Розширення сценаріїв | При виборі некоректних даних з'являється повідомлення про помилку |

Таблиця 1.9 - Варіант використання UC009

| | |
|------------|---|
| Назва | Перегляд візуалізації показників |
| Опис | Користувач може переглянути візуалізації аналітичних показників курсу валютної пари |
| Учасники | Користувач |
| Передумови | З'явилася потреба в графічному представленні результатів статистичного аналізу, необхідна для візуалізацій інформація наявна в базі даних сервісу |
| Постумови | Візуалізація зображена на екрані |

Продовження таблиці 1.9

| | |
|-------------------|---|
| Основний сценарій | Користувач перейшов з головної сторінки на сторінку візуалізацій, обрав валютну пару, період для аналізу, тип візуалізації, натиснув кнопку “Visualize” |
|-------------------|---|

Таблиця 1.10 - Варіант використання UC010

| | |
|-------------------|--|
| Назва | Експорт даних у файл |
| Опис | Користувач має можливість експортувати необхідну йому інформацію у файл |
| Учасники | Користувач |
| Передумови | З'явилася потреба розмістити дані в файлі на своєму комп'ютері, необхідна для завантаження інформація наявна в базі даних сервісу |
| Постумови | Файл з необхідною інформацією збережено на комп'ютері за вказаним шляхом |
| Основний сценарій | Користувач перейшов з головної сторінки на сторінку експорту курсу валют, обрав валютну пару, період для аналізу, формат файлу, шлях до директорії для збереження файлу, натиснув кнопку “Load info” |

Таблиця 1.11 - Варіант використання UC011

| | |
|------------|--|
| Назва | Розрахунок вартості конвертації |
| Опис | Користувач має можливість розрахувати вартість конвертації між різними валютами |
| Учасники | Користувач |
| Передумови | З'явилася потреба в розрахунку вартості конвертації між двома валютами, сервіс з інформацією для конвертації доступний |

Продовження таблиці 1.11

| | |
|----------------------|---|
| Постумови | Вартість конвертації відображена на екрані |
| Основний сценарій | Користувач перейшов з головної сторінки на сторінку розрахунку вартості конвертації, обрав валютну пару, кількість початкової валюти, натиснув кнопку "Convert" |
| Розширення сценаріїв | Користувач може змінювати одну з валютних пар або кількість валюти і переглядати результати різних розрахунків конвертації для порівняння, вони розмістяться один під одним |

Таблиця 1.12 - Варіант використання UC012

| | |
|-------------------|--|
| Назва | Вибір кількості валюти для розрахунку конвертації |
| Опис | Користувач може вводити необхідну для конвертації кількість валюти |
| Учасники | Користувач |
| Передумови | Користувач перейшов на сторінку розрахунку вартості конвертації |
| Постумови | Кількість валюти введена |
| Основний сценарій | Користувач переходить на сторінку розрахунку вартості конвертації і вводить кількість валюти для розрахунку вартості конвертації |

Таблиця 1.13 - Варіант використання UC013

| | |
|----------|---|
| Назва | Вибір формату файлу для збереження |
| Опис | Користувач може обрати один із запропонованих форматів файлів для експорту інформації |
| Учасники | Користувач |

Продовження таблиці 1.13

| | |
|-------------------|--|
| Передумови | Користувач перейшов на сторінку експорту інформації |
| Постумови | Формат збереження даних обраний |
| Основний сценарій | Користувач переходить на сторінку експорту інформації і обирає формат файлу для збереження |

Таблиця 1.14 - Варіант використання UC014

| | |
|-------------------|---|
| Назва | Вибір директорії для збереження файлу |
| Опис | Користувач може обрати директорію для збереження файлу на своєму присторі |
| Учасники | Користувач |
| Передумови | Користувач перейшов на сторінку експорту інформації |
| Постумови | Шлях до директорії для збереження даних обраний |
| Основний сценарій | Користувач переходить на сторінку експорту інформації і обирає директорію для збереження файлу на своєму присторі |

Функціональні вимоги додатку описано наступними таблицями.

Таблиця 1.15 - Опис функціональної вимоги REQ001

| | |
|-------|--|
| Номер | REQ001 |
| Назва | Ведення джерел інформації |
| Опис | Адміністратор може додавати і видаляти джерела інформації вручну |

Таблиця 1.16 - Опис функціональної вимоги REQ002

| | |
|-------|---|
| Номер | REQ002 |
| Назва | Ведення даних з різних сервісів |
| Опис | Адміністратор має готові скрипти і обгортки для API-інтерфейсів, написані мовою Python для часткової автоматизації завантаження інформації про курси валют до бази даних сервісу. Адміністратор може видаляти ці дані |

Таблиця 1.17 - Опис функціональної вимоги REQ003

| | |
|-------|---|
| Номер | REQ003 |
| Назва | Вибір періоду часу для аналізу |
| Опис | Користувач може вибирати початок і кінець періоду часу, для якого хоче отримати інформацію, статистичний аналіз чи переглянути візуалізацію |

Таблиця 1.18 - Опис функціональної вимоги REQ004

| | |
|-------|--|
| Номер | REQ004 |
| Назва | Вибір валютної пари для аналізу |
| Опис | Користувач може вибирати ліву і праву ногу валютної пари, для якої хоче отримати інформацію, статистичний аналіз чи переглянути візуалізацію |

Таблиця 1.19 - Опис функціональної вимоги REQ005

| | |
|-------|--|
| Номер | REQ005 |
| Назва | Вибір джерела інформації для аналізу |
| Опис | Користувач може джерело інформації, з якого будуть використані дані для подальшого аналізу |

Таблиця 1.20 - Опис функціональної вимоги REQ006

| | |
|-------|--|
| Номер | REQ006 |
| Назва | Перегляд аналітичних показників і статистичний аналіз даних |
| Опис | Після введення праметрів користувач має можливість переглянути результати статистичного і аналітичного аналізу курсу обраної валютної пари за обраний період |

Таблиця 1.21 - Опис функціональної вимоги REQ007

| | |
|-------|--|
| Номер | REQ007 |
| Назва | Візуалізації аналітичних показників та курсу валют |
| Опис | Після введення праметрів користувач має можливість переглянути візуалізацію аналітичних показників курсу обраної валютної пари за обраний період |

Таблиця 1.22 - Опис функціональної вимоги REQ008

| | |
|-------|---|
| Номер | REQ008 |
| Назва | Експорт даних у файл |
| Опис | Після введення праметрів користувач має можливість експортувати дані про курс обраної валютної пари за обраний період |

Таблиця 1.23 - Опис функціональної вимоги REQ009

| | |
|-------|--|
| Номер | REQ009 |
| Назва | Обчислення вартості конвертації між валютами |
| Опис | Після введення праметрів користувач має можливість отримати розрахунок вартості конвертації обраної кількості однієї валюти в іншу |

Таблиця 1.24 - Опис функціональної вимоги REQ010

| | |
|-------|---|
| Номер | REQ010 |
| Назва | Вибір формату файлу для експорту даних |
| Опис | На сторінці експорту даних до файлу користувач може обрати один із запропонованих форматів файлу для завантаження |

Таблиця 1.25 - Опис функціональної вимоги REQ011

| | |
|-------|--|
| Номер | REQ011 |
| Назва | Вибір директорії для експорту даних у файл |
| Опис | На сторінці експорту даних до файлу користувач може обрати директорію до якої потрібно зберегти файл (шлях до файлу) |

Таблиця 1.26 - Опис функціональної вимоги REQ012

| | |
|-------|--|
| Номер | REQ012 |
| Назва | Вибір типу візуалізації |
| Опис | На сторінці візуалізацій користувач може обрати необхідний йому тип візуалізації |

Таблиця 1.27 - Опис функціональної вимоги REQ013

| | |
|-------|--|
| Номер | REQ013 |
| Назва | Введення кількості валюти для конвертації |
| Опис | На сторінці конвертації валют користувач може обрати необхідну кількість валюти, для якої необхідно здійснити розрахунок |

Взаємозв'язки між вимогами клієнтського застосунку відображені на рисунку 1.1.

| | REQ001 Ведення джерел інформації. | REQ002 Ведення даних з різних сервісів. | REQ003 Вибір періоду часу для аналізу. | REQ004 Вибір валютної пари для аналізу. | REQ005 Вибір джерела інформації для аналізу. | REQ006 Перегляд аналітичних показників і статистичний аналіз даних. | REQ007 Візуалізації аналітичних показників та курсу валют. | REQ008 Експорт даних у файл. | REQ009 Обчислення вартості конвертації між валютами. | REQ010 Вибір формату файлу для експорту даних. файл. | REQ012 Вибір типу візуалізації. конвертації. |
|--|-----------------------------------|---|--|---|--|---|--|------------------------------|--|--|--|
| UC001 Ведення джерел інформації. | | | | | | | | | | | |
| UC002 Додавання джерел інформації. | | | | | | | | | | | |
| UC003 Видалення джерел інформації. | | | | | | | | | | | |
| UC004 Завантаження даних. | | | | | | | | | | | |
| UC005 Перегляд результатів статистичного аналізу. | | | | | | | | | | | |
| UC006 Вибір валютної пари. | | | | | | | | | | | |
| UC007 Вибір джерела інформації. | | | | | | | | | | | |
| UC008 Вибір періоду часу для аналізу. | | | | | | | | | | | |
| UC009 Перегляд візуалізації показників. | | | | | | | | | | | |
| UC010 Експорт даних у файл. | | | | | | | | | | | |
| UC011 Розрахунок вартості конвертації. | | | | | | | | | | | |
| UC012 Вибір кількості валюти для розрахунку конвертації. | | | | | | | | | | | |
| UC013 Вибір формату файлу для збереження. | | | | | | | | | | | |
| UC014 Вибір директорії для збереження файлу. | | | | | | | | | | | |

Рисунок 1.1 - Матриця залежності між вимогами застосунку і варіантами використання

1.4.2 Розроблення нефункціональних вимог

Вимоги до інтерфейсу:

- застосування має бути десктопним;
- необхідна можливість збереження даних в файли;
- сервіс має використовувати інформацію для аналізу із зовнішніх джерел.

Апаратні та програмні вимоги:

- сервіс повинен запускатися на пристроях з кольоровим дисплеєм (вимога для візуалізації);
- сервіс має працювати з операційними системами Ubuntu 16.04, Ubuntu 18.04.

Операційні вимоги:

- відновлюваність:
 - а) в разі збою необхідна можливість повторного виконання необхідної дії після перезавантаження сервісу;
 - б) в разі пошкодження бази даних необхідна можливість повторного завантаження даних з бекапу чи зі стороннього сервісу.
- ємність:
 - а) сервіс повинен мати можливість впровадження аналогічного до наявного джерела інформації (зовнішній сервіс) і розміщення даних з нього з розрахунку на те, що в середньому з одного джерела надходять погодинні історичні дані за 20 років про 60 валютних пар.
- збереження даних:
 - а) сервіс повинен зберігати інформацію про функціонування програми (результати виконання всіх запитів і взаємодії із зовнішніми джерелами) протягом одного року;
 - б) сервіс не повинен видаляти інформацію для аналізу з бази даних.
- продуктивність:
 - а) в сервісі кількість запитів на секунду не перевищує 100;
 - б) час відповіді системи на запит статистичного аналізу не більший ніж 1 секунда;
 - в) час відповіді сервісу на запит візуалізації не більший ніж 2 секунди.

1.4.3 Постановка комплексу завдань модулю

Розроблюване програмне забезпечення призначене для вирішення задачі агрегації та аналізу даних про курси валютних пар з різних сервісів.

Мета створення даної роботи – надання аналітичних послуг у сфері ринку іноземних валют, акумуляція в одному сервісі інформації з різних джерел і

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 31 |

проведення статистичного аналізу курсу валют на різних рівнях, спрощення процесу отримання та аналізу даних про конкретний стан ринку іноземних валют і зменшення витрат часу на пошук необхідної інформації.

Для досягнення мети даної роботи система повинна вирішувати наступні задачі:

- ведення даних з різних джерел;
- обчислення аналітичних показників;
- налаштування візуалізацій аналітичних та статистичних показників;
- експорт даних про курс валют;
- статистичний аналіз даних за обраний період;
- обчислення вартості конвертації між валютами.

Сервіс повинен працювати на пристроях зі встановленою операційною системою Ubuntu 16.04 і вище, а також на пристроях, де встановлені будь-які інші операційні системи сімейства Unix з можливими незначними порушеннями.

1.5 Математичне забезпечення

Розроблений сервіс надає можливість переглядати результати статистичного аналізу та візуалізації аналітичних показників. На сторінці аналізу курсу валют обчислюються такі величини як ануалізований коефіцієнт Шарпа (Annualized Sharpe Ratio), ануалізована відносна зміна курсу (Annualized Return), ануалізована волатильність (Annualized Volatility), просадка (Drawdown) та її параметри (момент і величина найбільшої просадки, тривалість просадки, тривалості періодів поглиблення просадки і виходу з неї). Також для обчислення просадки необхідно виконати проміжне обчислення такої величини як прибутковість (performance).

Вихідним параметром для обчислень є часовий ряд курсу валютної пари за певний період (масив R довжиною LENGTH, де LENGTH - кількість

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 32 |

елементів масиву). Алгоритм обчислення відносної зміни курсу, прибутковості та просадки наведений нижче:

– визначити відносну зміну курсу для кожної позначки часу:
 $RETURN[i] = (R[i] - R[i-1]) / R[i-1]$. Значення кожного $RETURN[i]$ належить проміжку $[-1, 1]$;

– визначити кумулятивну зміну курсу для кожної позначки часу:
 $CUM_RETURN[i] = (1 + RETURN[1]) * (1 + RETURN[2]) * \dots * (1 + RETURN[i]) - 1$. Значення кожного $CUM_RETURN[i]$ належить проміжку $[-1, 1]$. Отриманий масив є характеристикою відносної зміни курсу валюти за досліджуваний період відносно початкової позначки часу;

– визначити просадку для кожної позначки часу:
 а) визначити у скільки разів змінилося значення ціни валюти відносно ціни в початковий момент часу: $PERFORMANCE[i] = 1 + CUM_RETURN[i]$;

б) визначити $MAX_PERF = \max(PERFORMANCE[0, 1, \dots, i])$ - максимальне збільшення ціни;

в) визначити просадку для кожної позначки часу:
 $DRAWDOWN[i] = (MAX_PERF - PERFORMANCE[i]) / MAX_PERF$;

г) визначити максимальну просадку для кожної позначки часу:
 $MAXIMUM_DRAWDOWN[i] = \max(DRAWDOWN[0, 1, \dots, i])$.

Вихідними значеннями є два масиви, які містять значення поточної і максимальної просадки для кожної позначки часу ($DRAWDOWN$ і $MAXIMUM_DRAWDOWN$). Також результатами окремих кроків алгоритму є масиви значень відносної зміни курсу ($RETURN$) і прибутковості ($PERFORMANCE$) [5].

Коефіцієнт Шарпа [4] обчислюється за формулою

$$S = \frac{E[R - R_f]}{\sigma} \quad (1.1)$$

Чим більше значення цього показника, тим більш прибутковим і менш ризикованим вважається актив. Негативне значення показника сигналізує про те, що прибутковість активу низька і вкладення в такий актив принесе збитки.

Волатильність обчислюється як стандартне відхилення відносної зміни вартості активу.

Ануалізованим значенням величини є її середньорічне значення. Середньорічне значення обчислюється із врахуванням ще одного параметра – періоду часу (корінь від періоду часу в роках додається в знаменник), наприклад для значення волатильності σ_{SD} середньорічна волатильність обчислюється за формулою:

$$\sigma = \frac{\sigma_{SD}}{\sqrt{P}}, \quad (1.2)$$

де P – це період часу в роках [3].

Для побудови візуалізацій використовуються описані в цьому розділі величини та їх статистичні характеристики.

Currency Rate Chart – це графік залежності курсу валюти від часу. По осі x відмітка часу, по осі y значення курсу валюти для часової відмітки.

ACF Plot – графік, на якому зображена автокореляційна функція відносної зміни курсу валюти. Автокореляційна функція – це така залежність, яка виражається як кореляція функції з самою собою зміщеною на визначене значення (зсув). Ця візуалізація є корисною, адже з її допомогою легко помітити періодичність в часовому ряді величини.

Snail Trail Plot – графік, що показує залежність волатильності від відносної зміни ціни (return). За допомогою цієї візуалізації визначають як змінюється надлишковий прибуток і надлишковий ризик активу відносно цільової величини. Інтерпретується графік наступним чином:

- візуалізація ділиться на 4 квадранти;
- точки лівого верхнього квадранту відображають ті відмітки часу, коли прибуток перевершив теоретичне значення з відносно низьким ризиком;

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 34 |

- точки правого верхнього квадранту відображають ті відмітки часу, коли прибуток перевершив теоретичне значення з відносно високим ризиком;
- точки лівого нижнього квадранту відображають ті відмітки часу, коли збиток перевершив допустиме теоретичне значення з відносно низьким ризиком;
- точки лівого верхнього квадранту відображають ті відмітки часу, коли збиток перевершив допустиме теоретичне значення з відносно низьким ризиком.

ECDF Plot – зображення емпіричного кумулятивного розподілу відносних змін цін. По осі x return, по осі y – частка від періоду. Графік емпіричної функції розподілу інтерпретується аналогічно до інших функцій, таких як нормальний розподіл. За допомогою цієї візуалізації легко зрозуміти скільки відсотків досліджуваного періоду ціна змінювалася менше за x%.

QQ Plot – імовірнісний графік у математичній статистиці, який являє собою графічний метод для порівняння двох розподілів ймовірностей, ставлячи їх квантілі один проти одного. Якщо два порівнюваних розподілів ідентичні, графік Q-Q має вигляд лінії $y = x$. Q-Q графіки часто мають форму дуги або «S» форми, що показує, що один з розподілів більш асиметричний, ніж інший, або, що один з розподілів має важчий хвіст, ніж інший [6].

Drawdown Plot – графічне представлення зміни значення просадки та максимальної просадки з часом.

Приклади візуалізацій наведені в програмі та методиці тестування.

1.6 Висновки до розділу

В розділі 1 проведено:

- аналіз існуючих технічних засобів і відомих рішень;
- аналіз відомих програмних продуктів, їх переваг та недоліків;
- аналіз вимог до програмного забезпечення – розроблені функціональні та нефункціональні вимоги.

2 МОДЕЛЮВАННЯ ТА КОНСТРУЮВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Моделювання та аналіз програмного забезпечення

Додаток розроблено для операційної системи Ubuntu 16.04 і вище. Ubuntu – це найпопулярніший дистрибутив Linux, який надає мінімальний необхідний набір програм загального призначення. Дане програмне забезпечення справно працює і на інших операційних системах сімейства Unix в тому числі дистрибутивах Linux.

Для підтримки розширюваності коду необхідно виділити однакові ролі компонентів додатку і використовувати шаблони. Для графічного інтерфейсу створений клас Controller, який контролює процес перемикання між вікнами застосунку. Якщо в процесі розробки продукту виникне потреба в додаванні чи видалення одного чи кількох вікон інтерфейсу, потрібно лише відредагувати перелік доступних вікон в атрибутах цього класу, а реалізація графічного інтерфейсу кожного з вікон є незалежною від інших.

Також розширюваність коду забезпечена всередині кожної окремої частини застосунку. Наприклад, клас Charts реалізує вікно, в якому можна переглянути візуалізації показників у вигляді графіків. Тип візуалізації користувач обирає в процесі роботи, отже функція, що генерує графік повинна бути динамічним параметром. Власне, клас має статичний атрибут `vis_meth`, в якому перераховані типи візуалізацій і функції, що реалізують їх. Тепер при розширенні цієї частини функціоналу необхідно додати код нової функції в модуль візуалізацій і створити ще одне поле для відповідного типу візуалізації в цьому атрибуті. За вище описаним принципом побудовані всі частини даного застосунку, де результат залежить від вибору користувача в процесі роботи програми.

Застосунок було розроблено за допомогою мови Python 3.6, графічний інтерфейс користувача створений за допомогою додатку QtDesigner, який

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 36 |

дозволяє згенерувати програмний код на мові Python з використанням бібліотеки PyQt. Візуалізації (графічне представлення показників) реалізовані за допомогою модулю matplotlib, аналітична частина продукту реалізована за допомогою pandas, numpy, scipy, statsmodels, scipy. Структурна схема класів наведена графічних матеріалах. Опис класів сервісу наведено у таблиці 2.1, опис методів класів – у таблиці 2.2, опис статичних атрибутів класів сервісу – у таблиці 2.3, опис функцій модулів сервісу – у таблиці 2.4.

2.2 Архітектура програмного забезпечення

Діаграма класів програмного продукту наведена у графічному матеріалі. Детальний опис класів та функцій наведений нижче в таблицях.

Таблиця 2.1 - Опис класів сервісу

| Клас | Опис |
|--|--|
| RatesNotAvailableError (Exception) | Користувачська помилка, яка виникає у випадку, коли сервіс https://ratesapi.io не працює і не доступний для отримання інформації |
| DecimalFloatMismatchError (Exception) | Користувачська помилка, яка виникає при невірному форматі дробового числа в запиті до сервісу |
| Common | Базовий клас, що реалізує елементарні операції для роботи з сервісом https://ratesapi.io |
| CurrencyRates (Common) | Реалізує обгортку необхідних методів API-інтерфейсу для отримання інформації з сервісу https://ratesapi.io |
| CurrencyCodes | Клас для роботи із кодами (скороченими назвами) валют |

Продовження таблиці 2.1

| Клас | Опис |
|--|---|
| analytic_design.Ui_Form (object) | Реалізація графічних елементів сторінки призначеної для відображення результатів статистичного аналізу та аналітики |
| load_info_design.Ui_Form (object) | Реалізація графічних елементів сторінки призначеної для завантаження інформації з розробленого сервісу до файлу |
| charts_design.Ui_Form (object) | Реалізація графічних елементів сторінки призначеної для відображення візуалізацій різних аналітичних показників |
| converter_design.Ui_Form (object) | Реалізація графічних елементів сторінки призначеної для розрахунку вартості конвертації між валютами |
| main_design.Ui_MainWindow (object) | Реалізація графічних елементів сторінки призначеної для відображення результатів статистичного аналізу та аналітики |
| Controller | Реалізація обгортки |
| Analytic(QtWidgets.QWidget, analytic_design.Ui_Form) | Реалізація функціоналу сторінки призначеної для відображення результатів статистичного аналізу та аналітики |
| LoadInfo(QtWidgets.QWidget, load_info_design.Ui_Form) | Реалізація функціоналу сторінки призначеної для завантаження інформації з розробленого сервісу до файлу |
| Charts(QtWidgets.QWidget, charts_design.Ui_Form) | Реалізація функціоналу сторінки призначеної для відображення візуалізацій різних аналітичних показників |

Продовження таблиці 2.1

| Клас | Опис |
|--|---|
| Converter(QtWidgets.QWidget, converter_design.Ui_Form) | Реалізація функціоналу сторінки призначеної для розрахунку вартості конвертації між валютами |
| App(QtWidgets.QMainWindow, main_design.Ui_MainWindow) | Реалізація функціоналу головного вікна |
| MplCanvas(FigureCanvas) | Клас-обгортка для віджету FigureCanvas з модулю matplotlib.backends.backend_qt5agg |
| MplWidget(QWidget) | Реалізація віджету, що дозволяє зображати графіки, реалізовані за допомогою бібліотеки matplotlib у візуальному вікні PyQt4 |

Таблиця 2.2 - Опис методів класів сервісу

| Клас | Метод | Опис |
|--------|------------------|---|
| Common | __init__ | Конструктор. Ініціалізація початкових значень атрибутів Параметри: - force_decimal – boolean – вказує, чи приводити числа з плаваючою точкою до формату decimal, за замовчуванням False |
| Common | _source_url | Повертає рядок з базовою url-адресою сервісу з API-інтерфейсом |
| Common | _get_date_string | Визначення відмітки часу для отримання даних. Повертає рядок з часовою відміткою Параметри: - date_obj – datetime – відмітка часу для отримання даних |

Продовження таблиці 2.2

| Клас | Метод | Опис |
|-----------------------|-------------------|---|
| Common | _decode_rates | Перетворення отриманих з сервісу даних у визначений json-формат. Повертає отримані з сервісу дані Параметри: <ul style="list-style-type: none"> - response – дані з сервісу - use_decimal – boolean – вказує, чи приводити числа з плаваючою точкою до формату decimal, за замовчуванням False |
| Common | _get_decoded_rate | Отримання курсу однієї конкретної валюти Параметри: <ul style="list-style-type: none"> - response – дані з сервісу - dest_cur – str – друга валюта валютної пари (права нога) - use_decimal – boolean – вказує, чи приводити числа з плаваючою точкою до формату decimal, за замовчуванням False |
| CurrencyRates(Common) | get_rates | Отримання курсів вказаної валюти до всіх інших. Повертає отримані курси в json-форматі або повідомлення про помилку, якщо сервіс недоступний Параметри: <ul style="list-style-type: none"> - base_cur – str – базова валюта, та, для якої виконується запит - date_obj – datetime – відмітка часу для отримання даних, за замовчуванням None |

Продовження таблиці 2.2

| Клас | Метод | Опис |
|-----------------------|-----------|---|
| CurrencyRates(Common) | get_rate | Отримання курсу для двох конкретних валют на визначений момент часу. Повертає число – курс однієї валюти до іншої Параметри: <ul style="list-style-type: none"> - base_cur – str – базова валюта, ліва нога валютної пари - dest_cur – str – друга валюта валютної пари (права нога) - date_obj – datetime – відмітка часу для отримання даних, за замовчуванням None |
| CurrencyRates(Common) | convert | Конвертація заданої кількості однієї валюти в іншу. Повертає кількість другої валюти за курсом на вказаний момент часу Параметри: <ul style="list-style-type: none"> - base_cur – str – базова валюта, ліва нога валютної пари - dest_cur – str – друга валюта валютної пари (права нога) - amount – float – кількість грошей в одиницях першої валюти - date_obj – datetime – відмітка часу для отримання даних, за замовчуванням None |
| CurrencyCodes | _get_data | Отримання даних про валюту з її коду (скороченої назви). Повертає структуру dict з інформацією про валюту Параметри: <ul style="list-style-type: none"> - currency_code – str – код валюти |

Продовження таблиці 2.2

| Клас | Метод | Опис |
|-------------------------------|-------------------------------|--|
| CurrencyCodes | _get_data_from_symbol | Отримання даних про валюту з її назви. Повертає структуру dict з інформацією про валюту Параметри: - symbol – str – назва валюти |
| CurrencyCodes | get_symbol | Отримання скороченої назви валюти зі словника з інформацією про неї за заданим кодом валюти Параметри: - currency_code – str – код валюти |
| CurrencyCodes | get_currency_name | Отримання назви валюти зі словника з інформацією про неї за заданим кодом валюти Параметри: - currency_code – str – код валюти |
| CurrencyCodes | get_currency_code_from_symbol | Отримання коду валюти зі словника з інформацією про неї за заданою скороченою назвою Параметри: - symbol – str – скорочена назва валюти |
| main_design. Ui_MainWindow | setUpUi | Ініціалізація елементів головного вікна Параметри: - MainWindow – QtGui.QMainWindow – об'єкт зображуваного вікна |
| main_design. Ui_MainWindow | retranslateUi | Встановлення текстових значень для всіх елементів головного вікна Параметри: - MainWindow – QtGui.QMainWindow – об'єкт зображуваного вікна |

Продовження таблиці 2.2

| Клас | Метод | Опис |
|------------------------------|-----------------------------------|--|
| load_info_des ign.Ui_Form | setupUi | Ініціалізація елементів вікна завантаження даних Параметри: - Form– QtGui.Form– об’єкт поточного вікна |
| load_info_des ign.Ui_Form | retranslate Ui | Встановлення текстових значень для всіх елементів вікна завантаження інформації Параметри: - Form– QtGui.Form– об’єкт поточного вікна |
| converter_des ign.Ui_Form | setupUi | Ініціалізація елементів вікна розрахунку вартості конвертації Параметри: - Form– QtGui.Form– об’єкт поточного вікна |
| converter_des ign.Ui_Form | retranslate Ui | Встановлення текстових значень для всіх елементів вікна розрахунку вартості конвертації Параметри: - Form– QtGui.Form– об’єкт поточного вікна |
| charts_design .Ui_Form | setupUi | Ініціалізація елементів вікна візуалізації даних Параметри: - Form– QtGui.Form– об’єкт поточного вікна |
| charts_design .Ui_Form | retranslate Ui(self, Form): | Встановлення текстових значень для всіх елементів вікна візуалізації даних Параметри: - Form– QtGui.Form– об’єкт поточного вікна |
| analytic_desi gn.Ui_Form | setupUi | Ініціалізація елементів вікна відображення результатів аналітичного і статистичного аналізу Параметри: - Form– QtGui.Form– об’єкт поточного вікна |

Продовження таблиці 2.2

| Клас | Метод | Опис |
|--|-------------------|---|
| analytic_design.Ui_Form | retranslate Ui | Встановлення текстових значень для всіх елементів вікна відображення результатів аналітичного і статистичного аналізу Параметри: - Form– QtGui.Form– об'єкт поточного вікна |
| MplCanvas(FigureCanvas) | __init__ | Ініціалізація атрибутів екземпляру аналогічного FigureCanvas для використання з віджетів PyQt |
| MplWidget(QWidget) | __init__ | Ініціалізація атрибутів віджету для відображення візуалізації |
| Analytic(analytic_design.Ui_Form) | __init__ | Ініціалізація графічних елементів вікна і з'єднання функціональних елементів з відповідними методами. Встановлення зв'язку з базою даних |
| Analytic(analytic_design.Ui_Form) | switch | Створення сигналу про перехід до іншого вікна Параметри: - str_key – str – ідентифікатор вікна переходу |
| Analytic(analytic_design.Ui_Form) | get_params | Отримання введених користувачем параметрів зі змінюваних полів. Повертає структуру даних tuple – параметри для виконання подальшого аналізу |
| Analytic(QtWidgets.QWidget, analytic_design.Ui_Form) | calc_analytics | Обчислення аналітичних і статистичних показників. Повертає tuple, що складається з двох елементів типу pandas DataFrame: перший – ануалізовані величини, другий – таблиця просадок Параметри: - df – pandas DataFrame – набір даних для аналізу |

Продовження таблиці 2.2

| Клас | Метод | Опис |
|--|------------------|---|
| Analytic(QtWidgets.QWidget, analytic_design.Ui_Form) | make_result_text | Створення текстового повідомлення для виведення інформації користувачу. Повертає рядок, що містить інформацію про результати обчислень Параметри: - res – list – складається з двох елементів типу pandas DataFrame: перший – аналізовані величини, другий – таблиця просадок |
| Analytic(analytic_design.Ui_Form) | get_info | Запуск калькуляцій зі вказаними параметрами. Виводить результати на екран для перегляду |
| LoadInfo(load_info_design.Ui_Form) | __init__ | Ініціалізація графічних елементів вікна і з'єднання функціональних елементів з відповідними методами. Встановлення зв'язку з базою даних |
| LoadInfo(load_info_design.Ui_Form) | switch | Створення сигналу про перехід до іншого вікна Параметри: - str_key – str – ідентифікатор вікна переходу |
| LoadInfo(load_info_design.Ui_Form) | choose_directory | Вибір директорії в яку необхідно завантажити файл з інформацією |
| LoadInfo(load_info_design.Ui_Form) | get_params | Отримання введених користувачем параметрів зі змінюваних полів. Повертає структуру даних tuple – параметри для виконання подальшого аналізу |
| LoadInfo(load_info_design.Ui_Form) | load_info | Запуск завантаження інформації зі вказаними параметрами. Зберігає інформацію в обраній папці |

Продовження таблиці 2.2

| Клас | Метод | Опис |
|---|------------|---|
| Charts(charts_design.Ui_Form) | __init__ | Ініціалізація графічних елементів вікна і з'єднання функціональних елементів з відповідними методами. Встановлення зв'язку з базою даних |
| Charts(charts_design.Ui_Form) | switch | Створення сигналу про перехід до іншого вікна Параметри: - str_key – str – ідентифікатор вікна переходу |
| Charts(charts_design.Ui_Form) | get_params | Отримання введених користувачем параметрів зі змінюваних полів. Повертає структуру даних tuple – параметри для виконання подальшого аналізу |
| Charts(charts_design.Ui_Form) | plot_chart | Запуск зображення візуалізації зі вказаними параметрами. Зберігає інформацію в обраній папці |
| Converter(converter_design.Ui_Form) | __init__ | Ініціалізація графічних елементів вікна і з'єднання функціональних елементів з відповідними методами |
| Converter(converter_design.Ui_Form) | switch | Створення сигналу про перехід до іншого вікна Параметри: - str_key – str – ідентифікатор вікна переходу |
| Converter(converter_design.Ui_Form) | convert | Запуск обрахунку вартості конвертації однаєї валюти в іншу. Виводить результати на екран |
| App(QtWidgets.QMainWindow, main_design.Ui_MainWindow) | __init__ | Ініціалізація графічних елементів головного вікна і з'єднання функціональних елементів з відповідними методами |

Продовження таблиці 2.2

| Клас | Метод | Опис |
|--------------------------------|---------------|---|
| App(main_design.Ui_MainWindow) | switch | Створення сигналу про перехід до іншого вікна Параметри: - str_key – str – ідентифікатор вікна переходу |
| Controller | show_window | Відображення вікна за вказаним ідентифікатором Параметри: - str_key – str – ідентифікатор вікна переходу |
| Controller | buttonClicked | Обробка події натискання кнопки. Закриває попереднє вікно Параметри: - str_key – str – ідентифікатор вікна переходу |

Таблиця 2.3 - Опис статичних атрибутів класів сервісу

| Клас | Атрибут | опис |
|---|---------------|--|
| Analytic(QtWidgets.QWidget, analytic_design.Ui_Form) | switch_window | QtCore.pyqtSignal(str) сигнал, що викликає перехід вікна за вказаним ключем |
| Analytic(QtWidgets.QWidget, analytic_design.Ui_Form) | sources | Структура даних dict, формату ключ-значення, що пов'язує ідентифікатор джерела даних із ідентифікатором цього джерела даних в базі даних сервісу |
| LoadInfo(QtWidgets.QWidget, load_info_design.Ui_Form) | switch_window | QtCore.pyqtSignal(str) сигнал, що викликає перехід вікна за вказаним ключем |

Продовження таблиці 2.3

| Клас | Атрибут | опис |
|---|---------------|--|
| LoadInfo(QtWidgets.QWidget, load_info_design.Ui_Form) | sources | Структура даних dict, формату ключ-значення, що пов'язує ідентифікатор джерела даних із ідентифікатором цього джерела даних в базі даних сервісу |
| LoadInfo(QtWidgets.QWidget, load_info_design.Ui_Form) | f_formats | Структура даних dict, формату ключ-значення, що пов'язує ідентифікатор формату файлу із функцією, що зберігає файл у вказаний формат |
| Charts(QtWidgets.QWidget, charts_design.Ui_Form) | switch_window | QtCore.pyqtSignal(str) сигнал, що викликає перехід вікна за вказаним ключем |
| Charts(QtWidgets.QWidget, charts_design.Ui_Form) | sources | Структура даних dict, формату ключ-значення, що пов'язує ідентифікатор джерела даних із ідентифікатором цього джерела даних в базі даних сервісу |
| Charts(QtWidgets.QWidget, charts_design.Ui_Form) | vis_meth | Структура даних dict, формату ключ-значення, що пов'язує назву візуалізації із функцією, що її реалізує |
| Converter(QtWidgets.QWidget, converter_design.Ui_Form) | switch_window | QtCore.pyqtSignal(str) сигнал, що викликає перехід вікна за вказаним ключем |

Продовження таблиці 2.3

| Клас | Атрибут | опис |
|---|---------------|---|
| Converter(QtWidgets.QWidget, converter_design.Ui_Form) | sources | Структура даних dict, формату ключ-значення, що пов'язує ідентифікатор джерела даних із методом конвертації цього джерела |
| App(QtWidgets.QMainWindow, main_design.Ui_MainWindow) | switch_window | QtCore.pyqtSignal(str) сигнал, що викликає перехід вікна за вказаним ключем |
| Controller | bnt_window | Структура даних dict, формату ключ-значення, що пов'язує ідентифікатор вікна додатку з класом, що його реалізує |

Таблиця 2.4 - Опис функцій модулів сервісу

| Модуль | Функція | Опис |
|-----------------|--------------------------|---|
| module_analytic | calc_returns_performance | Обчислення відносних змін ціни валюти. Повертає набір даних з обчисленими полями “return”, “performance” Параметри: - df – pandas DataFrame – впорядкований в часі набір даних , що містить колонку “rate”, в якій знаходяться ціни валютної пари за обраний період |

Продовження таблиці 2.4

| Модуль | Функція | Опис |
|-----------------|-------------------|--|
| module_analytic | sharpe_ratio | Обчислення коефіцієнта Шарпа. Повертає число з плаваючою комою – значення коефіцієнта Параметри: <ul style="list-style-type: none"> - returns – pandas DataFrame – набір даних, що містить колонки “date” і “return” - start_time – datetime – початок періоду для аналізу, опціональний, за замовчанням None - end_time – datetime – кінець періоду для аналізу, опціональний, за замовчанням None |
| module_analytic | annualized_return | Обчислення значення ануалізованої зміни курсу валюти за обраний період. Повертає значення ануалізованого значення зміни ціни Параметри: <ul style="list-style-type: none"> - returns – pandas DataFrame – набір даних, що містить колонки “date” і “return” - start_time – datetime – початок періоду для аналізу, опціональний, за замовчанням None - end_time – datetime – кінець періоду для аналізу, опціональний, за замовчанням None |

Продовження таблиці 2.4

| Модуль | Функція | Опис |
|-----------------|--------------------|--|
| module_analytic | annualized_summary | Створення таблиці з ануалізованими значеннями зміни курсу ціни, стандартного відхилення ціни і коефіцієнту Шарпа за обраний період часу Параметри: <ul style="list-style-type: none"> - returns – pandas DataFrame – набір даних, що містить колонки “date” і “return” - start_time – datetime – початок періоду для аналізу, опціональний, за замовчанням None - end_time – datetime – кінець періоду для аналізу, опціональний, за замовчанням None |
| module_analytic | drawdowns_table | Створення таблиці з найбільшими просадками курсу валюти за обраний період Параметри: <ul style="list-style-type: none"> - returns – pandas DataFrame – набір даних, що містить колонки “date” і “return” - start_time – datetime – початок періоду для аналізу, опціональний, за замовчанням None - end_time – datetime – кінець періоду для аналізу, опціональний, за замовчанням None - top – int – кількість найбільших просадок для пошуку, опціональний, за замовчанням 10 |

Продовження таблиці 2.4

| Модуль | Функція | Опис |
|-----------|------------------|--|
| visualize | rate_plot | Функція, що додає до графічного віджету елементи візуалізації курсу валютної пари Параметри: <ul style="list-style-type: none"> - ax – matplotlib.axes.Axes – об'єкт, до якого функція додає необхідні елементи - df – pandas DataFrame – набір даних, що містить поля “date” і “rate” |
| visualize | acf_plot | Функція, що додає до графічного віджету елементи візуалізації автокореляції Параметри: <ul style="list-style-type: none"> - ax – matplotlib.axes.Axes – об'єкт, до якого функція додає необхідні елементи - df – pandas DataFrame – набір даних, що містить поля “date” і “return” |
| visualize | snail_trail_plot | Функція, що додає до графічного віджету елементи візуалізації Snail Trail Параметри: <ul style="list-style-type: none"> - ax – matplotlib.axes.Axes – об'єкт, до якого функція додає необхідні елементи - df – pandas DataFrame – набір даних, що містить поля “date” і “return”; - stepsize – integer – розмір кроку, за замовчуванням 12 |

Продовження таблиці 2.4

| Модуль | Функція | Опис |
|-----------|-----------|--|
| visualize | ecdf_plot | <p>Функція, що додає до графічного віджету елементи візуалізації емпіричного кумулятивного розподілу</p> <p>Параметри:</p> <ul style="list-style-type: none"> - <code>ax</code> – <code>matplotlib.axes.Axes</code> – об'єкт, до якого функція додає необхідні елементи - <code>df</code> – <code>pandas DataFrame</code> – набір даних, що містить поля “date” і “return” - <code>n_bins</code> – <code>integer</code> – кількість колонок гістограми, за замовчуванням 50 |
| visualize | qq_plot | <p>Функція, що додає до графічного віджету елементи візуалізації Q-Q графіка</p> <p>Параметри:</p> <ul style="list-style-type: none"> - <code>ax</code> – <code>matplotlib.axes.Axes</code> – об'єкт, до якого функція додає необхідні елементи - <code>df</code> – <code>pandas DataFrame</code> – набір даних, що містить поля “date” і “return” |
| visualize | dd_plot | <p>Функція, що додає до графічного віджету елементи візуалізації емпіричного кумулятивного розподілу</p> <p>Параметри:</p> <ul style="list-style-type: none"> - <code>ax</code> – <code>matplotlib.axes.Axes</code> – об'єкт, до якого функція додає необхідні елемент - <code>df</code> – <code>pandas DataFrame</code> – набір даних, що містить поля “date” і “performance” - <code>dd_period</code> – <code>integer</code> – період для обрахунку просадки 10 |

Продовження таблиці 2.4

| Модуль | Функція | Опис |
|-------------|-----------|---|
| Main (load) | save_xlsx | Завантаження інформації до файлу з розширенням “.xlsx” Параметри: <ul style="list-style-type: none"> - df – pandas DataFrame – інформація, яку потрібно записати до файлу - path – str – шлях до директорії, де необхідно зберегти файл |
| Main (load) | save_csv | Завантаження інформації до файлу з розширенням “.csv” Параметри: <ul style="list-style-type: none"> - df – pandas DataFrame – інформація, яку потрібно записати до файлу - path – str – шлях до директорії, де необхідно зберегти файл |
| Main (load) | save_html | Завантаження інформації до файлу з розширенням “.html” Параметри: <ul style="list-style-type: none"> - df – pandas DataFrame – інформація, яку потрібно записати до файлу - path – str – шлях до директорії, де необхідно зберегти файл |

2.3 Конструювання програмного забезпечення

Продукт має кілька варіантів використання: експорт даних до файлу, розрахунок вартості конвертації між валютами, статистичний аналіз цін і візуалізація аналітичних показників. Після запуску додатку обирається один із

вище описаних варіантів. Процес роботи кожного з них однаковий: ввести необхідні параметри, отримати результат.

Загальний алгоритм роботи з додатком:

- На головній сторінці необхідно натиснути на кнопку з назвою дії, яку потрібно виконати і програма перейде у відповідне вікно.
- У вікні розрахунку вартості конвертації необхідно обрати валюту для конвертації та ввести бажану кількість валюти. Після цього потрібно натиснути кнопку “Convert”. Програма виведе результат в текстове поле.
- У вікні завантаження даних необхідно обрати директорію для завантаження файлу, тип файлу, дві валюти (ліва і права ноги валютної пари), джерело даних, початок і кінець періоду часу для завантаження даних. Після цього потрібно натиснути кнопку “Load info”. Програма збереже файл у відповідності із вказаними параметрами.
- У вікні аналізу даних необхідно обрати дві валюти (ліва і права ноги валютної пари), джерело даних, початок і кінець періоду часу для завантаження даних. Після цього потрібно натиснути кнопку “Analyze”. Програма виведе результат в текстове поле.
- У вікні візуалізації даних необхідно обрати тип візуалізації, дві валюти (ліва і права ноги валютної пари), джерело даних, початок і кінець періоду часу для завантаження даних. Після цього потрібно натиснути кнопку “Plot”. Програма виведе результат в графічне поле.

Модель та нотація бізнес-процесів (BPMN) наведена на рисунку 2.1.

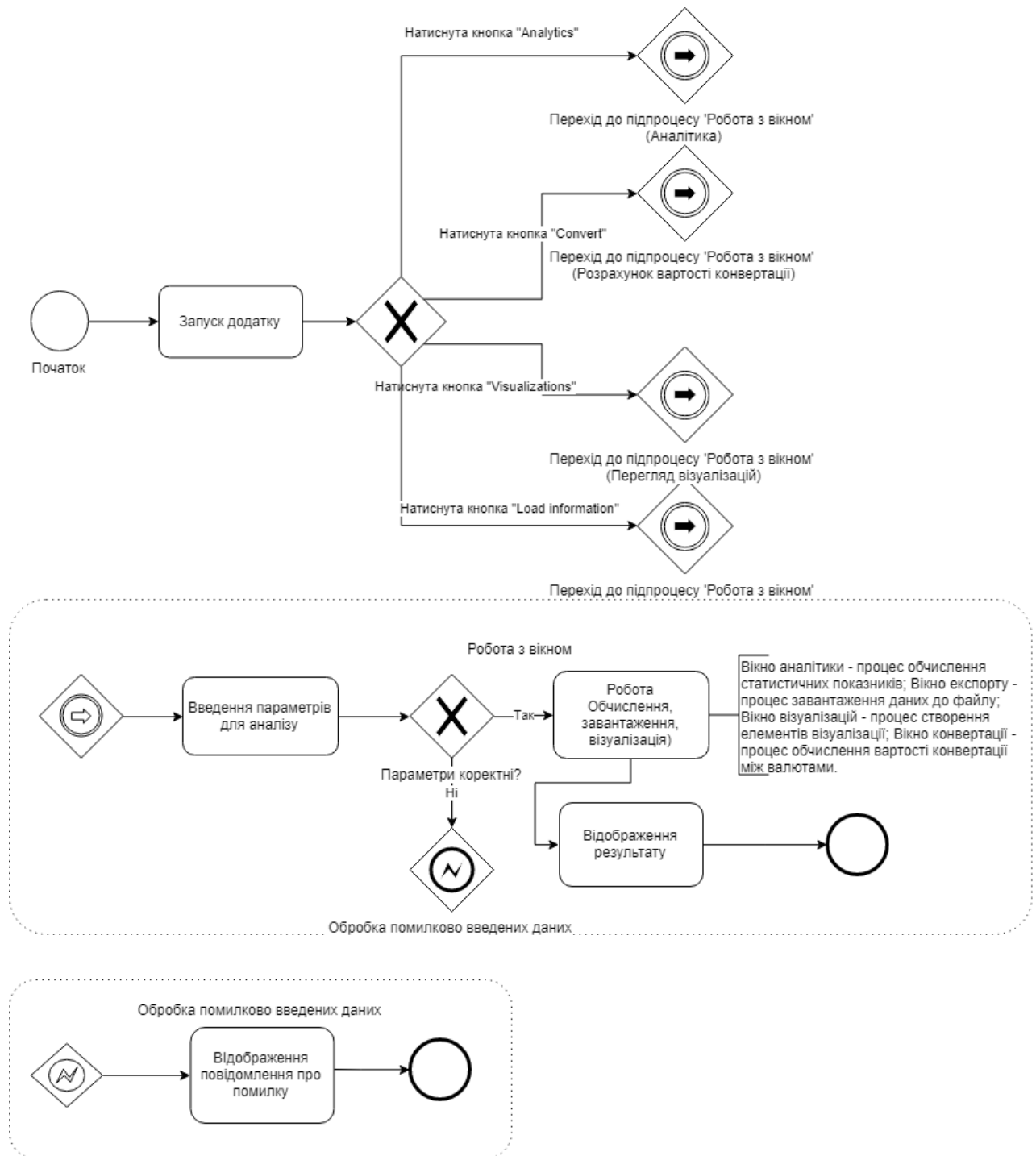


Рисунок 2.1 - Модель та нотація бізнес-процесів (BPMN) сервісу

2.4 Аналіз безпеки даних

Інформація про курси валютних пар потрапляє до бази даних сервісу за допомогою наданих різними сторонніми сервісами API-інтерфейсів і написаних на мові Python обгортки над ними. Основною гарантією безпечного з'єднання є

попередній відбір таких сервісів, що використовують надійні мережеві протоколи.

Як джерела інформації розробленого програмного продукту використовуються API-інтерфейси лише тих веб-сервісів, які використовують протоколи з шифруванням даних, наприклад HTTPS, TLS. Зокрема, сервіс ratesapi.io використовує протокол HTTPS.

HTTPS - схема URI, що синтаксично ідентична http: схемі, яка зазвичай використовується для доступу до ресурсів Інтернет. Використання https: URL вказує, що протокол HTTP має використовуватися, але з іншим портом за замовчуванням (443) і додатковим шаром шифрування/автентифікації між HTTP і TCP [7].

2.5 Висновки до розділу

В розділі 2 виконано моделювання та аналіз розробленого продукту, описано його архітектуру (специфікацію методів класів програмного забезпечення та функцій модулів) в діаграмі класів. Також проаналізовано та проілюстровано в BPMN діаграмі бізнес-процеси системи .

3 АНАЛІЗ ЯКОСТІ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

3.1 Аналіз якості ПЗ

Тестування є важливою і невід’ємною частиною розробки продуктів з надання послуг на ринку іноземних валют. Особливістю тестування застосунків призначених для використання в цій сфері є те, що на першому місці тут стоїть налагодження всіх складових продукту, де проводяться будь-які розрахунки і аналіз. Тобто основна мета - отримувати достовірні аналітичні і математичні результати на виході.

Важливою особливістю є те, що на ринку іноземних валют є дуже велика кількість валютних пар і значення будь-якого обчисленого показника для однієї пари буде відрізнятися від обчисленого значення для іншої. Також значення показника змінюється з часом і для однієї валютної пари. Це пояснюється тим, що різні валюти мають різну волатильність і різну вартість для кожного моменту часу, тому для тестування таких обчислень варто створювати еталонний набір даних і в першу чергу використовувати його для перевірки.

Такі додатки потрібно тестувати використовуючи дані з різних сервісів (джерел даних) для різних валют і проміжків часу. Різні джерела ринкових цін зазвичай містять відмінні від інших дані. Це пояснюється тим, що вони розташовані в різних країнах, а отже мають дещо відмінні курси валютних пар. Зазвичай дефекти в обчисленнях виявляються при використанні пошкоджених або штучно спотворених даних.

В таких продуктах візуалізації є вже продуктом різних аналітичних обчислень, тому варто перевіряти правильність відображення даних і відповідність цих даних параметрам, які ввів користувач.

Окремим етапом тестування є перевірка коректності відпрацювання всіх функцій інтерфейсу користувача.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КП.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 58 |

Точність розрахунків і аналітичних показників продукту в даній сфері є головною і ледь не єдиною мірою актуальності даного продукту в майбутньому і його конкурентоспроможності. На даний момент існує велика кількість сервісів і продуктів різного роду для надання послуг на ринку іноземних валют. Серед них онлайн сервіси, які збирають і агрегують дані з різних джерел, торгові платформи, аналітичні сервіси і т.д. Всі вони орієнтовані на різну аудиторію і мають різне призначення, але спільними для них є вимоги до точності, так як одиницею виміру є валюта, і будь-які похибки в роботі з грошима недопустимі. А якщо брати до уваги постійно зростаючу кількість конкурентів, то можна вважати, продукт з хибними обчисленнями мертвим.

3.2 Опис процесів тестування

Для тестування було використано пристрій зі встановленою операційною системою Ubuntu 16.04, під'єднаним до мережі інтернет і встановленими необхідними бібліотеками та пакетами, від яких залежить робота програми.

Для контролю якості застосунку в процесі тестування розроблені тест-кейси. Тести умовно можна розділити на загальні, функціональні і спеціальні тести.

Загальні тест-кейси для розробленого десктопного додатку описані в таблиці 3.1.

Таблиця 3.1 - Загальні тест-кейси

| Ідентифікатор | Опис |
|---------------|--|
| GC-1 | Перевірити процес запуску додатку |
| GC-2 | На кожній сторінці додатку натиснути кнопку повернення на головну сторінку ("Back to main page") |
| GC-3 | Для всіх сторінок додатку перевірити чи можливо його згорнути (Hide) |

Продовження таблиці 3.1

| Ідентифікатор | Опис |
|---------------|---|
| GC-4 | В разі успішного проходження GC-2 розгорнути вікно і перевірити збереження стану додатку після розгортання |
| GC-5 | Для всіх сторінок додатку перевірити чи можливий вихід з додатку при натисненні на хрестик в лівому верхньому куті екрану |
| GC-6 | З головної сторінки перейти на кожну з чотирьох функціональних сторінок додатку |
| GC-7 | Спробувати змінити розміри вікна для кожної сторінки |
| GC-8 | Для всіх випадючих полів на сторінках застосунку перевірити їх функціональність і можливість обирати параметри |
| GC-9 | Перевести пристрій в режим гібернації і перевірити роботу додатку після виходу з цього режиму. Те ж саме повторити для режиму сну |
| GC-10 | Перевірити роботу додатку при переході на задній план і виході з нього (наприклад випадки, коли під час роботи з продуктом на передньому плані з'являються сповіщення з інших додатків) |

Якість функціоналу є одним із найважливіших факторів, що визначає якість продукту. Тестування функціональності застосунку у відповідності до варіантів використання та розроблених функціональних вимог описано в таблиці 3.2.

Таблиця 3.2 - Функціональні тест-кейси

| Ідентифікатор | Опис |
|---------------|--|
| FC-1 | Перевірити можливість вибору початкової і кінцевої дати для подальшої роботи на всіх сторінках додатку |
| FC-2 | Перевірити можливість обрати ліву і праву ногу валютної пари при для подальшої роботи на всіх сторінках додатку |
| FC-3 | Перевірити можливість обирати джерело даних для подальшої роботи на всіх сторінках додатку |
| FC-4 | Перевірити можливість обирати формат файлу для збереження інформації в пам'яті пристрою |
| FC-5 | Перевірити, чи відповідає формат збережуваного файлу тому, який обрав користувач для всіх форматів файлів на сторінці експорту даних |
| FC-6 | Перевірити можливість обрати шлях до директорії для збереження файлу з інформацією у внутрішній пам'яті пристрою на сторінці експорту даних |
| FC-7 | Перевірити, чи зберігається файл на пристрої за шляхом, який обрав користувач на сторінці експорту даних |
| FC-8 | Перевірити коректність збереження інформації до внутрішньої пам'яті пристрою при введенні валідних параметрів на сторінці експорту даних |
| FC-9 | Перевірити збереження інформації до внутрішньої пам'яті пристрою при введенні некоректного шляху до директорії збереження на сторінці експорту даних |
| FC-10 | Перевірити збереження інформації до внутрішньої пам'яті пристрою при виборі однакової лівої і правої ноги валютної пари на сторінці експорту даних |

Продовження таблиці 3.2

| Ідентифікатор | Опис |
|---------------|---|
| FC-11 | Перевірити збереження інформації до внутрішньої пам'яті пристрою при виборі некоректних дат, що є початком і кінцем періоду для аналізу (дата початку більша за дату закінчення) на сторінці експорту даних |
| FC-12 | Перевірити можливість вибору типу візуалізації на сторінці графічного представлення аналітичних показників |
| FC-13 | Перевірити коректність відображення кожного типу візуалізацій при введенні валідних параметрів на сторінці графічного представлення аналітичних показників |
| FC-14 | Перевірити відображення кожного типу візуалізацій при виборі однакової лівої і правої ноги валютної пари на сторінці графічного представлення аналітичних показників |
| FC-15 | Перевірити відображення кожного типу візуалізацій при виборі некоректних дат, що є початком і кінцем періоду для аналізу (дата початку більша за дату закінчення) на сторінці графічного представлення аналітичних показників |
| FC-16 | Перевірити можливість введення кількості валюти для конвертації на сторінці розрахунку вартості конвертації |
| FC-17 | Перевірити можливість вибору від'ємного числа в полі кількості валюти для конвертації на сторінці розрахунку вартості конвертації |
| FC-18 | Перевірити розрахунок вартості конвертації (виведення інформації в текстове поле) при введенні валідних параметрів на сторінці розрахунку вартості конвертації |

Продовження таблиці 3.2

| Ідентифікатор | Опис |
|---------------|--|
| FC-19 | Перевірити розрахунок вартості конвертації (виведення інформації в текстове поле) при повторному запиті розрахунку змінивши одну з ніг валютної пари на сторінці розрахунку вартості конвертації. Повторити при зміні кількості валюти для конвертації |
| FC-20 | Перевірити розрахунок вартості конвертації (виведення інформації в текстове поле) при виборі однакової лівої і правої ноги валютної пари на сторінці розрахунку вартості конвертації |
| FC-21 | Перевірити розрахунок вартості конвертації (виведення інформації в текстове поле) при відсутньому підключенню до мережі інтернет на сторінці розрахунку вартості конвертації |
| FC-22 | Перевірити обчислення аналітичних показників та статистичний аналіз (виведення інформації в текстове поле) при введенні валідних параметрів на сторінці обчислення аналітичних показників та статистичного аналізу курсу валюти |
| FC-23 | Перевірити обчислення аналітичних показників та статистичний аналіз (виведення інформації в текстове поле) при повторному запиті обчислення та аналізу змінивши одну з ніг валютної пари на сторінці обчислення аналітичних показників та статистичного аналізу курсу валюти. Повторити при зміні періоду для аналізу курсу валюти |

Продовження таблиці 3.2

| Ідентифікатор | Опис |
|---------------|--|
| FC-24 | Перевірити обчислення аналітичних показників та статистичний аналіз (виведення інформації в текстове поле) при виборі однакової лівої і правої ноги валютної пари на сторінці обчислення аналітичних показників та статистичного аналізу курсу валюти |
| FC-25 | Перевірити обчислення аналітичних показників та статистичний аналіз (виведення інформації в текстове поле) при виборі некоректних дат, що є початком і кінцем періоду для аналізу (дата початку більша за дату закінчення) на сторінці обчислення аналітичних показників та статистичного аналізу курсу валюти |

В програмних продуктах, основним призначенням яких є надання аналітичних послуг на ринку іноземних валют дуже важливим показником якості є коректність і точність обчислюваних величин. Тому для тестування таких обрахунків необхідно створювати еталонні набори вхідних даних та обчислених для них результатів.

Тестування коректності обчислення аналітичних показників та спеціальні тест-кейси наведено в таблиці 3.3.

Таблиця 3.3 - Спеціальні тест-кейси

| Ідентифікатор | Опис |
|---------------|---|
| SC-1 | Перевірка точності обчислення технічних індикаторів, коефіцієнта Шарпа і ануалізованих аналітичних статистик при подачі на вхід коректних даних |

Продовження таблиці 3.3

| Ідентифікатор | Опис |
|---------------|--|
| SC-2 | Перевірка точності обчислення технічних індикаторів, коефіцієнта Шарпа і ануалізованих аналітичних статистик при подачі на вхід порожнього набору |
| SC-3 | Перевірка точності обчислення технічних індикаторів, коефіцієнта Шарпа і ануалізованих аналітичних статистик при подачі на вхід порожнього набору даних (курс валюти є константою) |
| SC-4 | Перевірка точності обчислення просадок за допомогою контрольного прикладу |
| SC-5 | Перевірка коректного розрахунку вартості конвертації |

Звіт тестування для всіх описаних вище тест-кейсів (загальні, функціональні, спеціальні) наведено в таблиці 3.4.

Таблиця 3.4 - Звіт тестування

| Ідентифікатор тесту | Очікуваний результат | Фактичний результат | Статус тесту |
|---------------------|---|---|--------------|
| GC-1 | Додаток запускається при подвійному натисканні на створену іконку | Додаток запускається при подвійному натисканні на створену іконку | Пройдено |
| GC-2 | Відбувся перехід з поточної сторінки на головну сторінку застосунку | Відбувся перехід з поточної сторінки на головну сторінку застосунку | Пройдено |
| GC-3 | Сторінка згорнута (прихована) | Сторінка згорнута (прихована) | Пройдено |

Продовження таблиці 3.4

| Ідентифікатор тесту | Очікуваний результат | Фактичний результат | Статус тесту |
|---------------------|---|---|--------------|
| GC-4 | Сторінка розгорнута, на екрані відображено все, що було відображено до згортання | Сторінка розгорнута, на екрані відображено все, що було відображено до згортання | Пройдено |
| GC-5 | Відбулося припинення роботи додатку | Відбулося припинення роботи додатку | Пройдено |
| GC-6 | Відбувся перехід на обрану сторінку додатку | Відбувся перехід на обрану сторінку додатку | Пройдено |
| GC-7 | Розмір вікна залишається незмінний | Розмір вікна залишається незмінний | Пройдено |
| GC-8 | В поля для введення можна ввести інформацію, всі випадючі поля розгортаються і дають можливість обирати серед наведених варіантів | В поля для введення можна ввести інформацію, всі випадючі поля розгортаються і дають можливість обирати серед наведених варіантів | Пройдено |
| GC-9 | Сторінка розгорнута, на екрані відображено все, що було відображено до переходу в режим гібернації/сну | Сторінка розгорнута, на екрані відображено все, що було відображено до переходу в режим гібернації/сну | Пройдено |

Продовження таблиці 3.4

| Ідентифікатор тесту | Очікуваний результат | Фактичний результат | Статус тесту |
|---------------------|--|--|--------------|
| GC-10 | При переході від стороннього додатку до роботи з розробленим застосунком сторінка розгорнута, на екрані відображено все, що було відображено до переходу застосунку на задній план | При переході від стороннього додатку до роботи з розробленим застосунком сторінка розгорнута, на екрані відображено все, що було відображено до переходу застосунку на задній план | Пройдено |
| FC-1 | Дату можна змінити натисканням клавіш “вверх”, “вниз”, натисканням на стрілочки біля відповідних полів або змінивши значення полів вручну на всіх сторінках додатку | Дату можна змінити натисканням клавіш “вверх”, “вниз”, натисканням на стрілочки біля відповідних полів або змінивши значення полів вручну на всіх сторінках додатку | Пройдено |
| FC-2 | Ліву і праву ногу валютної пари можна змінити натисканням клавіш “вверх”, “вниз”, або обрати з випадаючого меню на всіх сторінках додатку | Ліву і праву ногу валютної пари можна змінити натисканням клавіш “вверх”, “вниз”, або обрати з випадаючого меню на всіх сторінках додатку | Пройдено |

Продовження таблиці 3.4

| Ідентифікатор тесту | Очікуваний результат | Фактичний результат | Статус тесту |
|---------------------|---|---|--------------|
| FC-3 | На всіх сторінках додатку є можливість обирати джерело даних | На всіх сторінках додатку є можливість обирати джерело даних | Пройдено |
| FC-4 | Формат файлу для збереження інформації можна обрати натисканням клавіш “вверх”, “вниз”, або обрати з випадючого меню на сторінці експорту даних | Формат файлу для збереження інформації можна обрати натисканням клавіш “вверх”, “вниз”, або обрати з випадючого меню на сторінці експорту даних | Пройдено |
| FC-5 | При збереженні файлу його формат відповідає обраному формату | При збереженні файлу його формат відповідає обраному формату | Пройдено |
| FC-6 | Поле для вибору директорії збереження файлу працює | Поле для вибору директорії збереження файлу працює | Пройдено |
| FC-7 | Збереження файлу відбувається в обрану директорію | Збереження файлу відбувається в обрану директорію | Пройдено |
| FC-8 | Дані за обраний період для обраної валютної пари зберігаються у вказаному розширенні за обраним шляхом | Дані за обраний період для обраної валютної пари зберігаються у вказаному розширенні за обраним шляхом | Пройдено |

Продовження таблиці 3.4

| Ідентифікатор тесту | Очікуваний результат | Фактичний результат | Статус тесту |
|---------------------|--|--|--------------|
| FC-9 | Виведене повідомлення про помилку | Виведене повідомлення про помилку | Пройдено |
| FC-10 | Курс валюти є константою для обраного періоду. Дані збережені успішно | Курс валюти є константою для обраного періоду. Дані збережені успішно | Пройдено |
| FC-11 | Виведене повідомлення про помилку | Виведене повідомлення про помилку | Пройдено |
| FC-12 | Тип візуалізації можна змінити натисканням клавіш “вверх”, “вниз”, або обрати з випадаючого меню на сторінці графічного представлення аналітичних показників | Тип візуалізації можна змінити натисканням клавіш “вверх”, “вниз”, або обрати з випадаючого меню на сторінці графічного представлення аналітичних показників | Пройдено |
| FC-13 | Обрана візуалізація відображається у відповідності до вхідних параметрів | Обрана візуалізація відображається у відповідності до вхідних параметрів | Пройдено |
| FC-14 | Виведене повідомлення про те, що валюти однакові | Виведене повідомлення про те, що валюти однакові | Пройдено |

Продовження таблиці 3.4

| Ідентифікатор тесту | Очікуваний результат | Фактичний результат | Статус тесту |
|---------------------|---|---|--------------|
| FC-15 | Виведене повідомлення про помилку | Виведене повідомлення про помилку | Пройдено |
| FC-16 | На сторінці розрахунку вартості конвертації є змінюване поле для введення кількості валюти для розрахунку | На сторінці розрахунку вартості конвертації є змінюване поле для введення кількості валюти для розрахунку | Пройдено |
| FC-17 | При введенні від'ємного значення в поле, воно одразу замінюється на 0 | При введенні від'ємного значення в поле, воно одразу замінюється на 0 | Пройдено |
| FC-18 | Результати розрахунку виведені в текстове поле сторінки | Результати розрахунку виведені в текстове поле сторінки | Пройдено |
| FC-19 | Результати нового розрахунку виведені під попередніми результатами в тому ж текстовому полі | Результати нового розрахунку виведені під попередніми результатами в тому ж текстовому полі | Пройдено |
| FC-20 | Результатом обчислення є число рівне введеній кількості валюти для конвертації | Результатом обчислення є число рівне введеній кількості валюти для конвертації | Пройдено |

Продовження таблиці 3.4

| Ідентифікатор тесту | Очікуваний результат | Фактичний результат | Статус тесту |
|---------------------|---|---|--------------|
| FC-21 | Виведене повідомлення про помилку (так як немає доступу до сервісу) | Виведене повідомлення про помилку (так як немає доступу до сервісу) | Пройдено |
| FC-22 | Результати обчислення виведені в текстове поле сторінки | Результати обчислення виведені в текстове поле сторінки | Пройдено |
| FC-23 | Результати нового обчислення виведені під попередніми результатами в тому ж текстовому полі | Результати нового обчислення виведені під попередніми результатами в тому ж текстовому полі | Пройдено |
| FC-24 | Виведене повідомлення з попередженням про те, що валюти однакові | Виведене повідомлення з попередженням про те, що валюти однакові | Пройдено |
| FC-25 | Виведене повідомлення про помилку | Виведене повідомлення про помилку | Пройдено |
| SC-1 | Результат обчислення на тестовому наборі даних має відхилення не більше 0.1% від еталонного | Результат обчислення на тестовому наборі даних має відхилення не більше 0.1% від еталонного | Пройдено |
| SC-2 | Величини рівні 0 | Величини рівні 0 | Пройдено |
| SC-3 | Величини рівні 0 | Величини рівні 0 | Пройдено |

Продовження таблиці 3.4

| Ідентифікатор тесту | Очікуваний результат | Фактичний результат | Статус тесту |
|---------------------|---|---|--------------|
| SC-4 | Результат обчислення на тестовому наборі даних має відхилення не більше 0.1% від еталонного | Результат обчислення на тестовому наборі даних має відхилення не більше 0.1% від еталонного | Пройдено |
| SC-5 | Результат обчислення на тестовому наборі даних має відхилення не більше 0.1% від еталонного | Результат обчислення на тестовому наборі даних має відхилення не більше 0.1% від еталонного | Пройдено |

3.3 Опис контрольного прикладу

Розглянемо детальніше тест кейс FC-13: Перевірити коректність відображення кожного типу візуалізацій при введенні валідних параметрів на сторінці графічного представлення аналітичних показників. Перевірку наведено для графіку автокореляції.

| | |
|-----------------|---|
| Мета тесту | Перевірка коректності відтворення графіку автокореляції |
| Початковий стан | Відкритий сторінка візуалізації аналітичних показників. |
| Вхідні дані | <ul style="list-style-type: none"> – період часу для даних візуалізації: Start Date = 01.12.07, End Date = 01.01.08 – валютна пара: Left Currency = “EUR”, Right Currency = “USD” – тип візуалізації: “ACF Plot” |

| | |
|--|--|
| Схема проведення тесту | Ввести вхідні параметри у відповідні поля. Натиснути на кнопку "Plot" |
| Очікуваний результат | В правій частині екрану коректно відображена візуалізація автокореляції для періоду з 01.12.07 до 01.01.08. |
| Стан програмного продукту після проведення випробувань | В правій частині екрану коректно відображена візуалізація автокореляції для періоду з 01.12.07 до 01.01.08. (рисунок 3.1). |

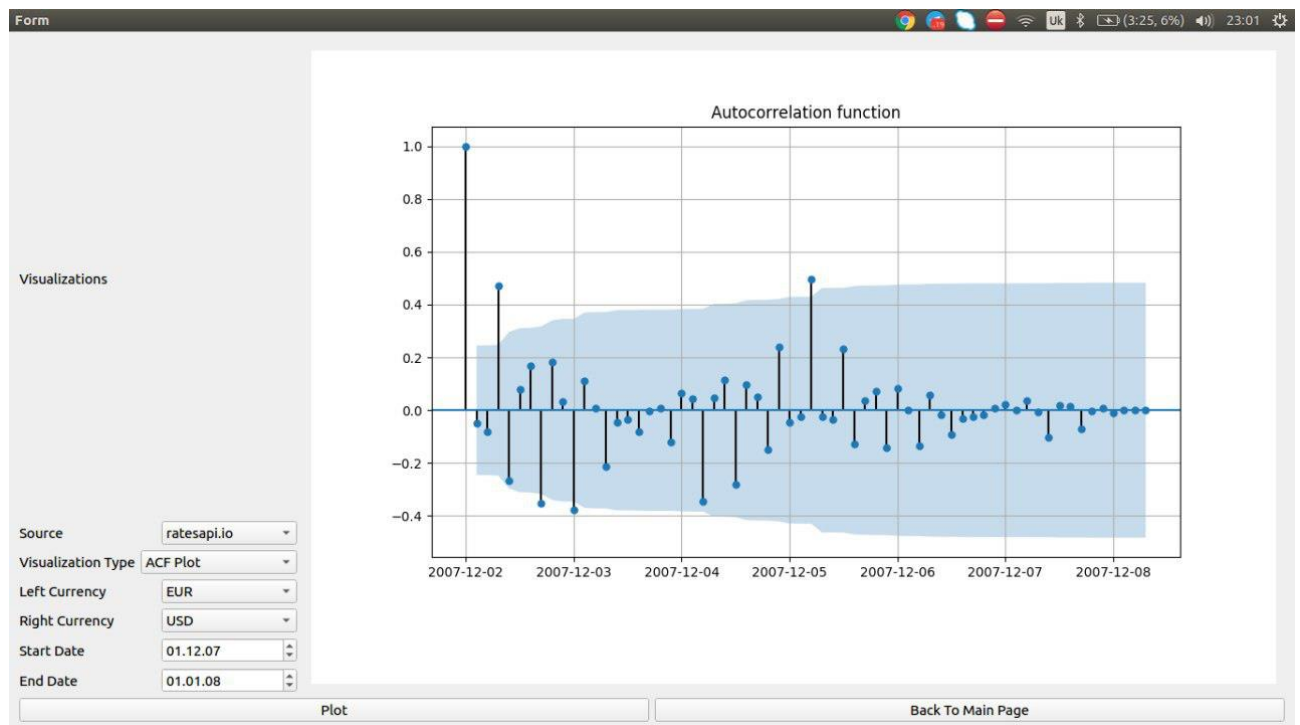


Рисунок 3.1 - Коректно відображена візуалізація автокореляції за обраний період

3.4 Висновки до розділу

В результаті тестування розробленого продукту розроблено та описано тест кейси для варіантів використання застосунку і з'ясовано, що якість продукту задовільна.

4 ВПРОВАДЖЕННЯ ТА СУПРОВІД ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Розгортання програмного забезпечення

Для роботи з розробленим програмним продуктом необхідно запустити виконуваний файл. Для операційних систем Ubuntu (версія 16.04 і вище) виконуваний файл додається до дипломного проекту і також доступний в репозиторії <https://github.com/AnnaYasenova/diplom>.

Для сімейства Unix операційних систем доданий файл може працювати некоректно. В цьому випадку можна запустити роботу сервісу з програмного коду попередньо скопіювавши текст програмного коду з репозиторію на комп'ютер або згенерувати виконуваний файл самостійно за допомогою модулю `pyinstaller` запустивши в командному рядку наступну команду `pyinstaller --onefile <шлях>/run.py`, де <шлях> - шлях до директорії, де потрібно розмістити виконуваний файл.

Для операційних систем сімейства Windows варто згенерувати виконуваний файл самостійно за допомогою модулю `PyWin32` або запустити роботу сервісу з програмного коду попередньо скопіювавши його на комп'ютер.

Для запуску додатку з програмного коду на пристрої, де запускається програма повинен бути встановлений Python версії 3.5 і вище з необхідними модулями, бібліотеками та розширеннями: `PyQt4` (або `PyQt5`), `pandas`, `numpy`, `scipy`, `matplotlib`, `statsmodels`, `glob`, `pymongo`.

Додаток призначений для використання на комп'ютерах та ноутбуках.

4.2 Робота з програмним забезпеченням

Інструкція для роботи з програмним забезпеченням наведена в керівництві користувача.

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 74 |

4.3 Висновки до розділу

В розділі 4 описано необхідні умови для розгортання розробленого програмного забезпечення та роботи з ним, розроблено керівництво користувача.

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| | | | | | | 75 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

ВИСНОВКИ

Виконання дипломного проекту проходило в декілька етапів, які оформлені в окремі розділи.

В розділі 1 проаналізовано проблематику надання аналітичних послуг на ринку іноземних валют та існуючих програмних рішень. Також проведено розробку функціональних та нефункціональних вимог до розроблюваного програмного забезпечення.

В розділі 2 проведено моделювання та конструювання розроблюваного сервісу, спроектована архітектура проекту, розроблені та описані класи згідно обраної архітектури спроектовано і розроблено десктопний додаток. Розроблений проект надає ряд переваг:

- можливість переглянути обчислені статистики і їх візуалізації в одному додатку;
- можливість обирати період для аналізу для порівняння результатів в різні проміжки часу;
- можливість аналізу, візуалізації, експорту і розрахунку конвертації з одного додатку;
- можливість додавати джерела даних для агрегації даних з них і подальшого порівняння результатів аналізу.

В розділі 3 проведено аналіз якості розробленого додатку і проведено його тестування. Було визначено, що додаток працює коректно і відповідає розробленим вимогам.

В розділі 4 описано процес розгортання додатку пристроях і необхідні умови для цього а також наведена інструкція користувача з описаними кроками використання.

Також розроблена проектна документація, BPMN додатку, схема варіантів використання і діаграма класів.

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 76 |

ПЕРЕЛІК ПОСИЛАНЬ

1. Ясенова А.В. Диверсифікація ризиків на ринку іноземних валют // II Всеукраїнська науково-практична конференція молодих вчених та студентів «Інформаційні системи та технології управління – ІСТУ-2019» Матеріали конференції. – 2019. – С.
2. Тренд (економіка) [Електронний ресурс]: (Стаття) / Wikipedia – Електрон. дан. (1 файл). – 2018. – Режим доступу: [https://ru.wikipedia.org/wiki/Тренд_\(економіка\)](https://ru.wikipedia.org/wiki/Тренд_(економіка)). – Назва з екрана.
3. Волатильність [Електронний ресурс]: (Стаття) / Wikipedia – Електрон. дан. (1 файл). – 2018. – Режим доступу: <https://ru.wikipedia.org/wiki/Волатильність>. – Назва з екрана.
4. Коэффициент Шарпа [Електронний ресурс]: (Стаття) / Wikipedia – Електрон. дан. (1 файл). – 2018. – Режим доступу: https://ru.wikipedia.org/wiki/Коэффициент_Шарпа. – Назва з екрана.
5. Ясенова А.В. Дослідження просадок курсу іноземних валют // Міжнародна науково-практична конференція «Економіка, фінанси, облік та право: стратегічні пріоритети розвитку в умовах глобалізації» Збірник тез доповідей. – 2019. – Ч. 2. – С. 22-24.
6. Графік Q-Q [Електронний ресурс]: (Стаття) / Wikipedia – Електрон. дан. (1 файл). – 2018. – Режим доступу: https://ru.wikipedia.org/wiki/Графік_Q-Q. – Назва з екрана.
7. HTTPS [Електронний ресурс]: (Стаття) / Wikipedia – Електрон. дан. (1 файл). – 2018. – Режим доступу: <https://ru.wikipedia.org/wiki/HTTPS>. – Назва з екрана.

ДОДАТОК А ТЕКСТ ПРОГРАМНОГО КОДУ

Тексти програмного коду
Сервіс аналізу ринку іноземних валют

(Найменування програми (документа))

DVD-R

(Вид носія даних)

13 арк, 2994 Кб

(Обсяг програми, арк., Кб)

Київ - 2019

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 78 |

module_analytic.py

```

import numpy as np
import pandas as pd
from datetime import timedelta

def calc_returns_performance(df):
    """
    :param df:
    :return:
    """
    df['return'] = (df['rate'] - df['rate'].shift()) / df['rate'].shift()
    df['performance'] = (df['return'] + 1).cumprod() - 1
    return df.fillna(0)

def sharpe_ratio(returns, start_time=None, end_time=None):
    """
    Calculate Sharpe Ratio for given timeseries
    :param returns - pandas DataFrame - contains column 'returns'
    :param start_time - datetime,timestamp - low bound of time, default None
    :param end_time - datetime,timestamp - high bound of time, default None
    :return value of Sharpe ratio
    """
    # returns = returns[pd.to_datetime(returns.date).dt.weekday < 5]
    if start_time:
        returns = returns[returns.date > start_time]
    if end_time:
        returns = returns[returns.date < end_time]

    std = float(returns['return'].std())
    avg_days_year = (returns.date.max() - returns.date.min()).days *
len(returns) / 365.25
    annual_multiplier = avg_days_year ** 0.5
    vol = annual_multiplier * std
    performance_annualized = ((1 + returns['return']) ** (avg_days_year * 1.0 /
len(returns))).product() - 1
    # TODO: if vol == 0
    sharpe = performance_annualized / vol
    return sharpe

def annualized_return(returns, start_time=None, end_time=None):
    """
    :param returns:
    :param start_time:
    :param end_time:
    :return:
    """
    if start_time:
        returns = returns[returns.date >= start_time]
    if end_time:
        returns = returns[returns.date <= end_time]

    avg_days_year = (returns.date.max() - returns.date.min()).days *
len(returns) / 365.25
    return ((1 + returns['return']) ** (avg_days_year * 1.0 /
len(returns))).product() - 1

def annualized_summary(returns, start_time=None, end_time=None):
    """

```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 79 |

Table of Annualized Return, Annualized Std Dev, and Annualized Sharpe.

```
:param returns - array-like - account's returns
:param start_time - datetime,timestamp - low bound of time, default None
:param end_time - datetime,timestamp - high bound of time, default None
"""
```

```
if start_time:
    returns = returns[returns.date > start_time]
if end_time:
    returns = returns[returns.date < end_time]
```

```
return pd.DataFrame({
    'Annualized Return': [annualized_return(returns, start_time=start_time,
end_time=end_time)],
    'Annualized Standard Deviation': [returns['return'].std() *
np.sqrt(252)],
    'Annualized Sharpe Ratio': [sharpe_ratio(returns, start_time=start_time,
end_time=end_time)]})
```

```
def drawdowns_table(returns, start_time=None, end_time=None, top=10):
```

"""

Creates table showing statistics for the worst drawdowns.

```
:param returns - array-like - account's returns
:param top - int - amount of worst drawdowns to show, default 10
"""
```

```
if start_time:
    returns = returns[returns.date > start_time - timedelta(days=31)]
if end_time:
    returns = returns[returns.date < end_time + timedelta(days=31)]
```

```
r_copy = returns[['date', 'performance']]
r_copy = r_copy.sort_values('date')
r_copy['modMax'] = r_copy['performance'].rolling(30, min_periods=1).max()
r_copy['modDD'] = 1 - r_copy['performance'].div(r_copy['modMax'])
groups = r_copy.groupby(r_copy['modMax'])
dd = groups['modMax', 'modDD'].apply(lambda g: g[g['modDD'] ==
g['modDD'].max()])
```

```
top10dd = dd.reset_index(drop=True).drop_duplicates().sort_values('modDD',
ascending=False).head(top)
```

```
top10dd = top10dd.merge(groups.min()[['date',
'modMax']]).reset_index(drop=True).rename(index=str,
```

```
columns={'date': 'From'}),
```

```
on='modMax', how='left')
```

```
top10dd = top10dd.merge(groups.max()[['date',
'modMax']]).reset_index(drop=True).rename(index=str,
```

```
columns={'date': 'To'}),
```

```
on='modMax', how='left')
```

```
top10dd = top10dd.merge(r_copy[['modDD', 'date']].rename(index=str,
```

```
columns={'date':
```

```
'Through'}),
```

```
on='modDD', how='left')
```

```
top10dd = (((top10dd.replace([np.inf, -np.inf], 0)[(top10dd['modMax'] != 0)
& (top10dd['modDD'] != 0)]
    ).sort_values('modMax', ascending=False)
    ).drop('modDD', axis=1)
```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| | | | | | | 80 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |


```

).rename(index=str, columns={'modMax': 'Depth'})

top10dd['Lenght'] = top10dd['To'] - top10dd['From']
top10dd['To Through'] = top10dd['Through'] - top10dd['From']
top10dd['Recovery'] = top10dd['To'] - top10dd['Through']
top10dd = top10dd.drop_duplicates('Depth')
return top10dd[top10dd.Lenght > '0 days']

```

visualize.py

```

from statsmodels.graphics.tsaplots import plot_acf
from scipy import stats

```

```

def rate_plot(ax, df):
    """

```

```

:param ax:
:param df:
:return:
"""

```

```

df.set_index('date')['rate'].plot(ax=ax)
ax.grid(True)
ax.set_title('Currency rate')

```

```

def acf_plot(ax, df):
    """

```

```

Plot autocorrelation chart.
See:

```

```

http://qualityamerica.com/LSS-Knowledge-Center/qualityimprovementtools/interpreting\_an\_autocorrelation\_chart.php
:param ax:
:param df:
:return:
"""

```

```

# TODO: if too big dataset
plot_acf(df.set_index('date')['return'], ax=ax)
ax.set_xticklabels(df['date'].dt.date, fontdict=None, minor=False) #[i for i
in range(len(df))],
ax.grid(True)
ax.set_title('Autocorrelation function')

```

```

def snail_trail_plot(ax, df, stepsize = 12): # TODO step parameter
    """

```

```

Plot SnailTrail chart that shows rolling calculations of annualized
return and annualized standard deviation have proceeded through time.
See:

```

```

https://www.rdocumentation.org/packages/PerformanceAnalytics/versions/1.5.2/topics/chart.SnailTrail
https://support.lonsec.com.au/hc/en-us/articles/204799965-Quant-charts-explained-Snail-Trail
:param ax:
:param df:
:param stepsize:
:return:
"""

```

```

deviations = []
for ind, row in df.iterrows():
    deviations.append(df[df['date'] <= row['date']]['return'].std())
df['Volatility'] = deviations
df.plot.scatter(x='Volatility', y='return', ax=ax,
                title='Snail Trail Chart')

x_line = df['Volatility'].values[::stepsize]
y_line = df['return'].values[::stepsize]
ax.plot(x_line, y_line, color='red')
ax.grid(True)
ax.set_xlabel('Returns')
ax.set_ylabel('Cumulative frequency')

def ecdf_plot(ax, df, n_bins=50): #TODO
    """
    Plot SnailTrail chart that shows rolling calculations of annualized
    return and annualized standard deviation have proceeded through time.
    See:

    https://www.rdocumentation.org/packages/PerformanceAnalytics/versions/1.5.2/topics/chart.SnailTrail
    https://support.lonsec.com.au/hc/en-us/articles/204799965-Quant-charts-explained-Snail-Trail
    :param ax:
    :param df:
    :param n_bins:
    :return:
    """

    # n, bins, patches =
    ax.hist(df['return'].values, n_bins, linewidth=3, histtype='step',
            cumulative=True)

    ax.grid(True)
    ax.set_title('Empirical cumulative distribution function')
    ax.set_xlabel('Returns')
    ax.set_ylabel('Cumulative frequency')

def qq_plot(ax, df):
    """
    Plot the return data against any theoretical distribution.
    See:

    https://www.rdocumentation.org/packages/PerformanceAnalytics/versions/1.5.2/topics/chart.QQPlot
    https://en.wikipedia.org/wiki/Q%E2%80%93Q_plot
    :param ax:
    :param df:
    :return:
    """

    stats.probplot(df['return'], sparams=(2.5,), plot=ax)
    ax.grid(True)
    ax.set_title('Q-Q plot\n(Return data against theoretical distribution)')
    ax.set_xlabel('Theoretical quantiles')
    ax.set_ylabel('Sample quantiles')

def dd_plot(ax, df, dd_period=10):
    """

```

A time series chart demonstrating drawdowns from peak equity attained through time, calculated from periodic returns.
See:

<https://www.rdocumentation.org/packages/PerformanceAnalytics/versions/1.5.2/topics/chart.Drawdown>

```
:param ax:
:param df:
:param dd_period:
:return:
"""

Roll_Max = df.performance.rolling(dd_period, min_periods=1).max()
Daily_Drawdown = df.performance / Roll_Max - 1.0
Daily_Drawdown[Daily_Drawdown > 0] = 0
Max_Daily_Drawdown = Daily_Drawdown.rolling(dd_period, min_periods=1).min()

ret_plot = df[['performance', 'date']].copy()
ret_plot['drawdown'] = Daily_Drawdown
ret_plot['max_drawdown'] = Max_Daily_Drawdown
ret_plot = ret_plot.dropna()

ret_plot.set_index('date')['drawdown'].plot.area(color='#9b59b6', alpha=.6,
ax=ax)
ret_plot.set_index('date')['max_drawdown'].plot(color='red', lw=2, alpha=.8,
ax=ax)

ax.grid(True)
ax.set_title('Drawdown')
ax.set_xlabel('Date')
ax.set_ylabel('Value')
```

run.py

```
import sys
import pandas as pd
from PyQt5 import QtWidgets, QtCore
import main_design, analytic_design, load_info_design, charts_design,
converter_design

from ratesapi_io.rates_io_api import CurrencyRates

from visualize import acf_plot, rate_plot, snail_trail_plot, ecdf_plot, qq_plot,
dd_plot

import module_analytic as m_a

import pymongo
db_name = 'diplom'

class Analytic(QtWidgets.QWidget, analytic_design.Ui_Form):

    switch_window = QtCore.pyqtSignal(str)
    sources = {
        'ratesapi.io': 'rateapiio'
    }
    def __init__(self):
        super().__init__()
        self.setupUi(self)
```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| | | | | | | 83 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

```

self.btn_back_to_main.clicked.connect(lambda: self.switch('main'))
self.btn_get_info.clicked.connect(self.get_info)

self.db = pymongo.MongoClient()[db_name]

def switch(self, str_key):
    self.switch_window.emit(str_key)

def get_params(self):
    return str(self.choose_source.currentText()),
    str(self.choose_left_cur.currentText()),\
        str(self.choose_right_cur.currentText()),
    pd.Timestamp(self.start_date.date().toPyDate()),\
        pd.Timestamp(self.end_date.date().toPyDate())

def calc_analytics(self, df):
    return m_a.annualized_summary(df), m_a.drawdowns_table(df)

def make_input_text(self, source, left_cur, right_cur, start_date,
end_date):
    return ('=====\nInformation source: ' +
source +
        '\nCurrency pair to analyze: ' + left_cur + right_cur +
'\nPeriod to ' +
        'analyze: ' + str(start_date.date()) + ' - ' +
str(end_date.date()))

def make_result_text(self, res):
    """
    :param res:
    :return:
    """
    dd_tbl = res[1]
    if len(dd_tbl) > 0:
        pass
    return '=====\nSUMMARY\n' +
res[0].transpose().to_string(header=False)

def get_info(self):
    source, left_cur, right_cur, start_date, end_date = self.get_params()

    collection = self.db[self.sources[source] + '_' + left_cur.lower()]
    # TODO: check dates
    df = pd.DataFrame(list(collection.find(
        {'date': {'$gte': start_date, '$lte': end_date}},
        {'_id': 0, 'date': 1, right_cur: 1})))
    ).rename(index=str, columns={right_cur: 'rate'})
    df = m_a.calc_returns_performance(df)

    self.textBrowser.append(self.make_input_text(source, left_cur, right_cur
, start_date, end_date))

self.textBrowser.append(self.make_result_text(self.calc_analytics(df.copy())))

def save_xlsx(df, path):
    df.to_excel(path, engine='xlsxwriter', index=False)

def save_csv(df, path):
    df.to_csv(path, index=False)

```

```
def save_html(df, path):
    df.to_html(path, index=False)

class LoadInfo(QtWidgets.QWidget, load_info_design.Ui_Form):

    switch_window = QtCore.pyqtSignal(str)

    sources = {
        'ratesapi.io': 'rateapiio'
    }

    f_formats = {
        '.xlsx': save_xlsx,
        '.csv': save_csv,
        '.html': save_html
    }

    def __init__(self):

        super().__init__()
        self.setupUi(self)
        self.btn_back_to_main.clicked.connect(lambda: self.switch('main'))
        self.btn_get_info.clicked.connect(self.load_info)
        self.btn_choose_dir.clicked.connect(self.choose_directory)

        self.db = pymongo.MongoClient()[db_name]
        self.path_to_save = ''

    def switch(self, str_key):
        self.switch_window.emit(str_key)

    def choose_directory(self):
        self.path_to_save = QtWidgets.QFileDialog.getExistingDirectory(self,
        'Select directory to save file')

    def get_params(self):
        return str(self.choose_source.currentText()),
        str(self.choose_left_cur.currentText()), \
        str(self.choose_right_cur.currentText()),
        pd.Timestamp(self.start_date.date().toPyDate()), \
        pd.Timestamp(self.end_date.date().toPyDate()),
        str(self.box_choose_file_format.currentText())

    def load_info(self):
        source, left_cur, right_cur, start_date, end_date, file_format =
        self.get_params()

        path_to_save = left_cur.lower() + '_' + right_cur.lower() + '_' + \
        str(start_date.date()) + '_' + str(end_date.date()) +
        file_format

        path_to_save = self.path_to_save + '/' + path_to_save if
        self.path_to_save != '' else path_to_save

        collection = self.db[self.sources[source]+'_'+left_cur.lower()]
        df = pd.DataFrame(list(collection.find({'date': {'$gte': start_date,
        '$lte': end_date}}),
```

```

        {'_id': 0, 'date': 1, right_cur:
1))))
        self.f_formats[file_format](df, path_to_save)

class Charts(QtWidgets.QWidget, charts_design.Ui_Form):

    switch_window = QtCore.pyqtSignal(str)

    sources = {
        'ratesapi.io': 'rateapiio'
    }

    vis_meth = {
        "Currency Rate Chart": rate_plot,
        "ACF Plot": acf_plot,
        "Snail Trail Plot": snail_trail_plot,
        "ECDF Plot": ecdf_plot,
        "QQ Plot": qq_plot,
        "Drawdown Plot": dd_plot
    }

    def __init__(self):

        super().__init__()
        self.setupUi(self)
        self.btn_back_to_main.clicked.connect(lambda: self.switch('main'))
        self.btn_get_info.clicked.connect(self.plot_chart)

        self.db = pymongo.MongoClient()[db_name]

    def switch(self, str_key):
        self.switch_window.emit(str_key)

    def get_params(self):
        return str(self.choose_source.currentText()), \
            str(self.choose_vis_type.currentText()), \
            str(self.choose_left_cur.currentText()), \
            str(self.choose_right_cur.currentText()), \
            pd.Timestamp(self.start_date.date().toPyDate()), \
            pd.Timestamp(self.end_date.date().toPyDate())

    def plot_chart(self):
        source, vis_type, left_cur, right_cur, start_date, end_date = \
            self.get_params()

        collection = self.db[self.sources[source] + '_' + left_cur.lower()]
        df = pd.DataFrame(list(collection.find(
            {'date': {'$gte': start_date, '$lte': end_date}},
            {'_id': 0, 'date': 1, right_cur: 1})))
        ).rename(index=str, columns={right_cur: 'rate'})
        df = m_a.calc_returns_performance(df)

        self.MplWidget.canvas.ax.clear()
        self.vis_meth[vis_type](self.MplWidget.canvas.ax, df)
        self.MplWidget.canvas.draw()

class Converter(QtWidgets.QWidget, converter_design.Ui_Form):

```

```

switch_window = QtCore.pyqtSignal(str)
sources = {
    'ratesapi.io': CurrencyRates().convert
}

def __init__(self):

    super().__init__()
    self.setupUi(self)
    self.btn_back_to_main.clicked.connect(lambda: self.switch('main'))
    self.btn_convert.clicked.connect(self.convert)

def switch(self, str_key):
    self.switch_window.emit(str_key)

def convert(self):
    source = str(self.choose_source.currentText())
    left_cur = str(self.choose_from.currentText())
    right_cur = str(self.choose_to.currentText())
    amount = str(self.amount.value())
    conv_res = self.sources[source](left_cur, right_cur, float(amount))
    self.text_browse_res.append('*****\n'
                                'Currency converter.\n'
                                '{} {} is equal to {} {}.'.format(amount,
                                                                    left_cur, round(conv_res, 3), right_cur))

class App(QtWidgets.QMainWindow, main_design.Ui_MainWindow):

    switch_window = QtCore.pyqtSignal(str)

    def __init__(self):

        super().__init__()
        self.setupUi(self)
        self.btn_analytic.clicked.connect(lambda: self.switch('analytic'))
        self.btn_charts.clicked.connect(lambda: self.switch('chart'))
        self.btn_load.clicked.connect(lambda: self.switch('load'))
        self.btn_convert.clicked.connect(lambda: self.switch('convert'))

    def switch(self, str_key):
        self.switch_window.emit(str_key)

class Controller:

    bnt_window = {
        'analytic': Analytic,
        'chart': Charts,
        'load': LoadInfo,
        'convert': Converter,
        'main': App
    }

    def __init__(self):
        pass

    def show_window(self, str_key):
        self.window = self.bnt_window[str_key]()
        self.window.show()
        self.window.switch_window.connect(self.buttonClicked)

```

```

def buttonClicked(self, str_key):
    self.window.close()
    self.show_window(str_key)

def main():
    app = QtWidgets.QApplication(sys.argv)
    controller = Controller()
    controller.show_window('main')
    sys.exit(app.exec_())

if __name__ == '__main__':
    main()

```

ratesapi_io.rates_io_api.py

```

import os
from decimal import Decimal
import simplejson as json
import requests

class RatesNotAvailableError(Exception):
    """
    Custome Exception when https://ratesapi.io is Down and not available for
    currency rates
    """
    pass

class DecimalFloatMismatchError(Exception):
    """
    A float has been supplied when force_decimal was set to True
    """
    pass

class Common:

    def __init__(self, force_decimal=False):
        self._force_decimal = force_decimal

    def _source_url(self):
        return "https://api.ratesapi.io/api/"

    def _get_date_string(self, date_obj):
        if date_obj is None:
            return 'latest'
        date_str = date_obj.strftime('%Y-%m-%d')
        return date_str

    def _decode_rates(self, response, use_decimal=False):
        if self._force_decimal or use_decimal:
            decoded_data = json.loads(response.text,
            use_decimal=True).get('rates', {})
        else:
            decoded_data = response.json().get('rates', {})
        return decoded_data

```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| | | | | | | 88 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |


```

def _get_decoded_rate(self, response, dest_cur, use_decimal=False):
    return self._decode_rates(response,
use_decimal=use_decimal).get(dest_cur, None)

class CurrencyRates(Common):

    def get_rates(self, base_cur, date_obj=None):
        date_str = self._get_date_string(date_obj)
        payload = {'base': base_cur, 'rtype': 'fpy'}
        source_url = self._source_url() + date_str
        response = requests.get(source_url, params=payload)
        if response.status_code == 200:
            rates = self._decode_rates(response)
            return rates
        raise RatesNotAvailableError("Currency Rates Source Not Ready")

    def get_rate(self, base_cur, dest_cur, date_obj=None):
        if base_cur == dest_cur:
            if self._force_decimal:
                return Decimal(1)
            return 1.
        date_str = self._get_date_string(date_obj)
        payload = {'base': base_cur, 'symbols': dest_cur, 'rtype': 'fpy'}
        source_url = self._source_url() + date_str
        response = requests.get(source_url, params=payload)
        if response.status_code == 200:
            rate = self._get_decoded_rate(response, dest_cur)
            if not rate:
                raise RatesNotAvailableError("Currency Rate {0} => {1} not
available for Date {2}".format(
                    base_cur, dest_cur, date_str))
            return rate
        raise RatesNotAvailableError("Currency Rates Source Not Ready")

    def convert(self, base_cur, dest_cur, amount, date_obj=None):
        if isinstance(amount, Decimal):
            use_decimal = True
        else:
            use_decimal = self._force_decimal

        if base_cur == dest_cur: # Return same amount if both base_cur,
dest_cur are same
            if use_decimal:
                return Decimal(amount)
            return float(amount)

        date_str = self._get_date_string(date_obj)
        payload = {'base': base_cur, 'symbols': dest_cur, 'rtype': 'fpy'}
        source_url = self._source_url() + date_str
        response = requests.get(source_url, params=payload)
        if response.status_code == 200:
            rate = self._get_decoded_rate(response, dest_cur,
use_decimal=use_decimal)
            if not rate:
                raise RatesNotAvailableError("Currency {0} => {1} rate not
available for Date {2}.".format(
                    source_url, dest_cur, date_str))
            try:
                converted_amount = rate * amount
                return converted_amount

```

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.02.81 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 89 |

```

        except TypeError:
            raise DecimalFloatMismatchError("convert requires amount
parameter is of type Decimal when force_decimal=True")
            raise RatesNotAvailableError("Currency Rates Source Not Ready")

_CURRENCY_FORMATTER = CurrencyRates()

get_rates = _CURRENCY_FORMATTER.get_rates
get_rate = _CURRENCY_FORMATTER.get_rate
convert = _CURRENCY_FORMATTER.convert

class CurrencyCodes:

    def __init__(self):
        pass

    def _get_data(self, currency_code):
        file_path = os.path.dirname(os.path.abspath(__file__))
        with open(file_path+'/raw_data/currencies.json') as f:
            currency_data = json.loads(f.read())
            currency_dict = next((item for item in currency_data if item["cc"] ==
currency_code), None)
            return currency_dict

    def _get_data_from_symbol(self, symbol):
        file_path = os.path.dirname(os.path.abspath(__file__))
        with open(file_path + '/raw_data/currencies.json') as f:
            currency_data = json.loads(f.read())
            currency_dict = next((item for item in currency_data if item["symbol"]
== symbol), None)
            return currency_dict

    def get_symbol(self, currency_code):
        currency_dict = self._get_data(currency_code)
        if currency_dict:
            return currency_dict.get('symbol')
        return None

    def get_currency_name(self, currency_code):
        currency_dict = self._get_data(currency_code)
        if currency_dict:
            return currency_dict.get('name')
        return None

    def get_currency_code_from_symbol(self, symbol):
        currency_dict = self._get_data_from_symbol(symbol)
        if currency_dict:
            return currency_dict.get('cc')
        return None

_CURRENCY_CODES = CurrencyCodes()

get_symbol = _CURRENCY_CODES.get_symbol
get_currency_name = _CURRENCY_CODES.get_currency_name
get_currency_code_from_symbol = _CURRENCY_CODES.get_currency_code_from_symbol

```

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Сервіс аналізу ринку іноземних валют

Технічне завдання

КПІ.ІІ-5225.045490.03.91

“ПОГОДЖЕНО”

Керівник проекту:

_____ О.А.Халус

Нормоконтроль:

_____ К.І. Ліщук

Виконавець:

_____ А.В.Ясенова

Київ – 2019 року

ЗМІСТ

| | | |
|----------|--|-----------|
| 1 | НАЙМЕНУВАННЯ ТА ГАЛУЗЬ ЗАСТОСУВАННЯ..... | 3 |
| 2 | ПІДСТАВА ДЛЯ РОЗРОБКИ..... | 4 |
| 3 | ПРИЗНАЧЕННЯ РОЗРОБКИ | 5 |
| 4 | ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ | 6 |
| 4.1 | Вимоги до функціональних характеристик..... | 6 |
| 4.2 | Вимоги до надійності | 6 |
| 4.3 | Умови експлуатації | 6 |
| 4.4 | Вимоги до складу і параметрів технічних засобів | 7 |
| 4.5 | Вимоги до інформаційної та програмної сумісності | 8 |
| 4.6 | Вимоги до маркування та пакування..... | 8 |
| 4.7 | Вимоги до транспортування та зберігання | 8 |
| 4.8 | Спеціальні вимоги..... | 8 |
| 5 | ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ | 9 |
| 6 | СТАДІЇ І ЕТАПИ РОЗРОБКИ..... | 10 |
| 7 | ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ..... | 12 |

1 НАЙМЕНУВАННЯ ТА ГАЛУЗЬ ЗАСТОСУВАННЯ

Назва розробки: Сервіс аналізу ринку іноземних валют

Галузь застосування: Наведене технічне завдання поширюється на розробку програмного забезпечення «Сервіс аналізу ринку іноземних валют» КПІ.ІП-5225.045490, котра використовується для швидкого отримання інформації про курс валютної пари з одного додатку для різних джерел інформації, обчислення вартості конвертації однієї валюти в іншу і отримання статистичного і аналітичного аналізу курсу валюти за обраний період та призначена для трейдерів які торгують на одній чи кількох платформах будь-якою кількістю валютних пар.

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.03.91 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 3 |

2 ПІДСТАВА ДЛЯ РОЗРОБКИ

Підставою для розробки Сервісу аналізу ринку іноземних валют є завдання на дипломне проектування, затверджене кафедрою автоматизованих систем обробки інформації і управління Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського» (КПІ ім.Ігоря Сікорського).

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.03.91 | Арк. |
| | | | | | | 4 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

3 ПРИЗНАЧЕННЯ РОЗРОБКИ

Розробка призначена для трейдерів які торгують на одній чи кількох платформах будь-якою кількістю валютних пар.

Метою розробки є надання аналітичних послуг у сфері ринку іноземних валют, акумуляція в одному сервісі інформації з різних джерел і проведення статистичного аналізу курсу валют на різних рівнях, спрощення процесу отримання та аналізу даних про конкретний стан ринку іноземних валют і зменшення витрат часу на пошук необхідної інформації.

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.03.91 | Арк. |
| | | | | | | 5 |
| Змн. | Арк. | № докум. | Підпис | Дата | | |

4 ВИМОГИ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Вимоги до функціональних характеристик

4.1.1 Програмне забезпечення повинно забезпечувати виконання наступних основних функцій:

4.1.1.1 Для користувача:

- завантаження курсу обраної валютної пари;
- обчислення вартості конвертації між валютами;
- статистичний аналіз курсу валютної пари за вказаний період;
- налаштування візуалізації необхідних статистик.

4.1.2 Розробку виконати на платформі Ubuntu 16.04.

4.1.3 Додаткові вимоги:

– для завантаження курсу, статистичного аналізу та візуалізацій надати можливість обирати період для аналізу.

4.2 Вимоги до надійності

Надійне функціонування програми повинно бути забезпечене виконанням Замовником організації безперебійного струмопостачання технічних засобів.

4.2.1 Передбачити контроль введення інформації.

4.2.2 Передбачити захист від некоректних дій користувача.

4.2.3 Забезпечити цілісність інформації в базі даних.

4.3 Умови експлуатації

Кліматичні умови експлуатації, при яких повинні забезпечуватися задані характеристики, повинні задовольняти вимогам, що заявлені до технічних засобів зокрема до умов їх експлуатації.

4.3.1 Умови експлуатації згідно СанПін 2.2.2.542 – 96.

4.3.2 Обслуговування

Даний продукт потребує систематичного обслуговування, а саме додавання нової інформації про курс валют до бази даних.

4.3.3 Обслуговуючий персонал

Для обслуговування достатньо одного адміністратора або додаткового програмного продукту (скрипта), який буде виконувати обслуговування автоматично.

4.4 Вимоги до складу і параметрів технічних засобів

4.4.1 Програмне забезпечення повинно функціонувати на ІВМ-сумісних персональних комп'ютерах.

4.4.2 Мінімальна конфігурація технічних засобів:

Тип процесору x86_64.

Об'єм ОЗП 1024Мб.

Достатній об'єм жорсткого диску 1 Гб

Клавіатура

4.5 Вимоги до інформаційної та програмної сумісності

4.5.1 Програмне забезпечення повинно працювати під управлінням операційних систем сімейства Unix.

4.5.2 Вхідні дані повинні бути представлені в наступному форматі випадального вікна з можливими варіантами значень того чи іншого параметра.

4.5.3 Результати повинні бути представлені в наступному форматі: файл обраного формату (.csv, .xlsx або .html) при завантаженні інформації, текстовий опис результату при розрахунку вартості конвертації та аналізі курсу валюти, графічне зображення при візуалізації статистик.

4.5.4 Програмне забезпечення повинно працювати з обраними джерелами інформації за допомогою обгортки для API інтерфейсів наданих розробниками або написаними власноруч.

На комп'ютері, де має бути встановлений додаток, необхідно наступне програмне забезпечення:

- операційна система Windows (7, 8, 10) або будь - яка *NIX сумісна;
- середовище Python 3 (версія не менше 3.5.1);
- СКБД MongoDB (версія не менше 3.2).

4.6 Вимоги до маркування та пакування

Вимоги до маркування та пакування не пред'являються.

4.7 Вимоги до транспортування та зберігання

Вимоги до транспортування та зберігання не пред'являються.

4.8 Спеціальні вимоги

Згенерувати установчу версію програмного забезпечення.

5 ВИМОГИ ДО ПРОГРАМНОЇ ДОКУМЕНТАЦІЇ

5.1 Програмні модулі, котрі розробляються, повинні бути задокументовані, тобто тексти програм повинні містити всі необхідні коментарі.

5.2 Програмне забезпечення повинно мати довідникову систему

5.3 У склад супроводжувальної документації повинні входити наступні документи:

5.3.1 Пояснювальна записка не менше ніж на 100 аркушах формату А4 (без додатків 5.3.2 - 5.3.6).

5.3.2 Технічне завдання.

5.3.3 Керівництво користувача.

5.3.4 Програма та методика тестування

5.4 Графічна частина повинна бути виконана на листах формату А3, котрі включаються до “Графічних матеріалів”:

5.4.1 Схема структура варіантів використання

5.4.2 Схема структурна класів програмного забезпечення

5.4.3 Креслення вигляду екранних форм.

6 СТАДІЇ І ЕТАПИ РОЗРОБКИ

| № | Назва етапу | Строк | Звітність |
|----|--|------------|--|
| 1. | Вивчення рекомендованої літератури | 19.03.2019 | |
| 2. | Аналіз існуючих методів розв'язання задачі | 26.03.2019 | |
| 3. | Постановка та формалізація задачі | 26.03.2019 | Технічне завдання |
| 4. | Аналіз вимог до програмного забезпечення | 02.04.2019 | Схема структура варіантів використання, вимоги до програмного забезпечення |
| 5. | Алгоритмізація задачі | 02.04.2019 | |
| 6. | Моделювання програмного забезпечення | 09.04.2019 | |
| 7. | Обґрунтування використовуваних технічних засобів | 16.04.2019 | |
| 8. | Розробка архітектури програмного забезпечення | 23.04.2019 | Схема структурна класів програмного забезпечення |
| 9. | Розробка програмного забезпечення | 30.04.2019 | Тексти програмного забезпечення |

| | | | |
|-----|----------------------------------|------------|---------------------------------|
| 10. | Налагодження програми | 07.05.2019 | Програма та методика тестування |
| 11. | Виконання графічних документів | 14.05.2019 | Графічний матеріал проекту |
| 12. | Оформлення пояснювальної записки | 21.05.2019 | Пояснювальна записка проекту |
| 13. | Подання ДП на попередній захист | 28.05.2019 | |
| 14. | Подання ДП рецензенту | 03.05.2019 | |
| 15. | Подання ДП на основний захист | 08.06.2019 | |

7 ПОРЯДОК КОНТРОЛЮ ТА ПРИЙМАННЯ

7.1 Види випробувань

Тестування розробленого програмного продукту виконується відповідно до “Програми та методики тестування”.

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.03.91 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 12 |

Факультет інформатики та обчислювальної техніки
Кафедра автоматизованих систем обробки інформації і управління

“ЗАТВЕРДЖЕНО”

В.о. завідувача кафедри

_____ О.А. Павлов

“ ____ ” _____ 2019 р.

Сервіс аналізу ринку іноземних валют

Програма та методика тестування

КПІ.ІІ-5225.045490.04.51

“ПОГОДЖЕНО”

Керівник проекту:

_____ О.А.Халус

Нормоконтроль:

_____ К.І. Ліщук

Виконавець:

_____ А.В.Ясенова

Київ – 2019 року

ЗМІСТ

| | |
|---------------------------------------|---|
| 1. ОБ'ЄКТ ВИПРОБУВАНЬ..... | 3 |
| 2. МЕТА ТЕСТУВАННЯ | 3 |
| 3. МЕТОДИ ТЕСТУВАННЯ..... | 3 |
| 4. ЗАСОБИ ТА ПОРЯДОК ТЕСТУВАННЯ | 4 |

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.04.51 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 2 |

1 ОБ'ЄКТ ВИПРОБУВАНЬ

Сервіс аналізу ринку іноземних валют, який являє собою десктопний застосунок, створений з використанням мови програмування Python з використанням оболонки PyQt бібліотеки Qt (інтерфейс), та ряду бібліотек, таких як numpy, pandas, scipy, matplotlib.

2 МЕТА ТЕСТУВАННЯ

У процесі тестування має бути перевірено наступне:

- функціональна працездатність елементів сторінок сервісу;
- коректність відображення візуалізацій;
- коректність обчислень та розрахунків аналітичної частини сервісу;
- відповідність формату вихідних результатів на всіх сторінках сервісу обраним вхідним параметрам;
- зручність роботи з сервісом;
- відповідність дизайну вимогам Технічного завдання.

3 МЕТОДИ ТЕСТУВАННЯ

Тестування виконується методом Gray Box Testing. Перевіряється як код, так і безпосередньо програмний продукт на відповідність функціональним вимогам. Тестування відбувається на рівні модульного тестування та системного тестування. За ознакою позитивності проведені тести поділяються на позитивне і негативне тестування, за ступенем автоматизації на напівавтоматизоване і ручне тестування.

Використовуються наступні методи:

- дослідницьке тестування;
- функціональне тестування;
- тестування інтерфейсу користувача.

4 ЗАСОБИ ТА ПОРЯДОК ТЕСТУВАННЯ

Тестування проводиться за допомогою засобів бібліотеки Unittest.

Працездатність сервісу перевіряється шляхом:

- динамічного ручного тестування — введенням граничних та недопустимих значень в поля, які можна редагувати;
- динамічного ручного тестування на відповідність функціональним вимогам;
- статичного тестування коду;
- тестування коректності обчислень за допомогою підготованих наборів даних;
- тестування стабільності роботи при різних умовах;
- тестування зручності використання;
- тестування інтерфейсу.

ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ
Кафедра автоматизованих систем обробки інформації і управління

«ЗАТВЕРДЖЕНО»

В.о. завідувача кафедри

_____ О.А.Павлов
(підпис) (ініціали, прізвище)

“ ” _____ 2019 р.

«Сервіс аналізу ринку іноземних валют»

Керівництво користувача

КП.ІП-5225.045490.05.34

“ПОГОДЖЕНО”

Керівник проекту:

_____ Халус О.А.

Нормоконтроль:

_____ Ліщук К.І.

Виконавці:

_____ Ясенова А.В.

Київ – 2019 року

Користувач запускає додаток подвійним натисканням на іконку виконуваного файлу. Відкривається головне вікно програми (рисунок 1), на якому зображене головне меню.

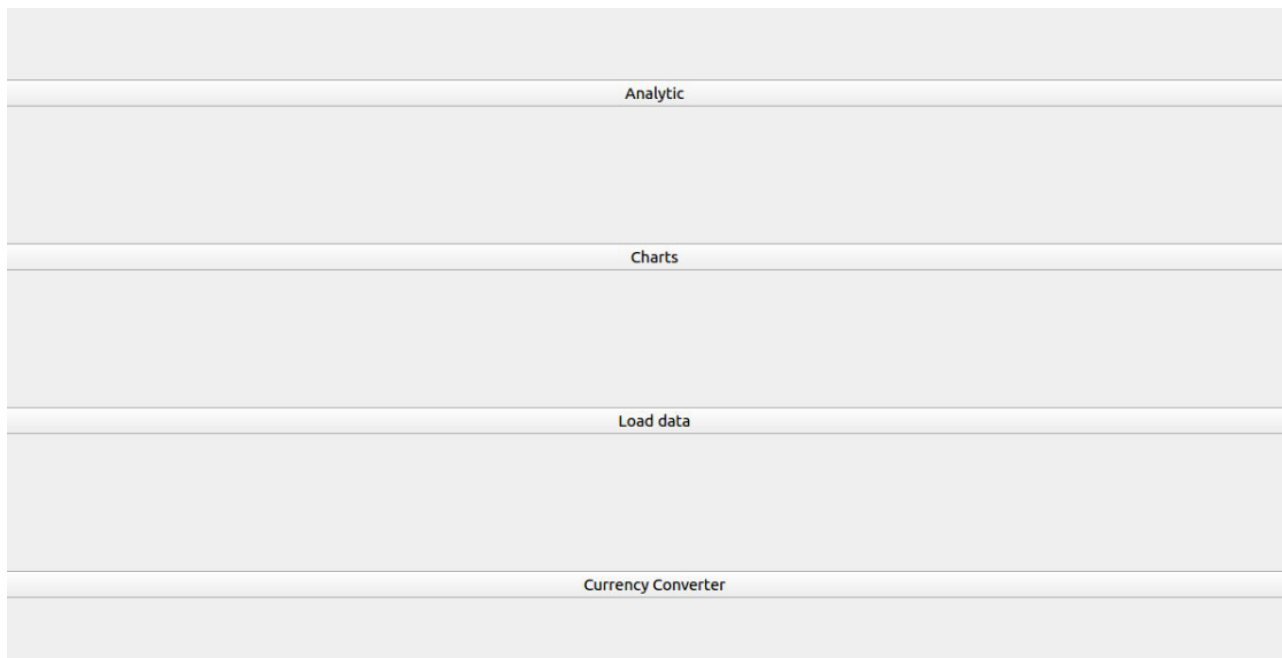


Рисунок 1 - Головна сторінка програми

Управління в головному меню виконується за допомогою натискання лівою кнопкою миші на кнопки переходу до наступних сторінок.

При натисканні на кнопку “Analytic” користувач переходить на сторінку статистичного аналізу та аналітичної оцінки курсу валюти за вказаний період (рисунок 2).

В лівому нижньому куті знаходяться поля вибору вхідних параметрів для аналізу. Необхідно обрати джерело даних в полі “Source”, ліву і праву ногу валютної пари в полях “Left Currency” і “Right Currency” відповідно, початок і кінець періоду часу для аналізу в полях “Start Date” і “End Date” відповідно.

Для отримання результату необхідно натиснути кнопку “Get information”.

Якщо обраний некоректний період часу для аналізу (значення поля “Start Date” більше або рівне значенню поля “End Date”) в поле результату буде виведене відповідне повідомлення.

Рисунок 2 - Сторінка статистичного аналізу та аналітичної оцінки курсу
валюти

Якщо ліва і права нога валютної пари однакові в поле результату буде виведене відповідне повідомлення.

Якщо неможливо виконати якесь із обчислень (наприклад при відсутності даних в БД) також виведене відповідне повідомлення.

Якщо обрані параметри коректні, то в текстовому полі відобразяться вхідні параметри та результати аналізу даних враховуючи обрані обмеження (рисунок 3).

На даній сторінці є можливість змінити один чи кілька вхідних параметрів і результати для нового набору параметрів відобразяться нижче в текстовому полі.

Для повернення на головну сторінку сервісу (рисунок 1) необхідно натиснути на кнопку “Back to Main Page”.

| | | | | | | |
|------|------|----------|--------|------|-------------------------|------|
| | | | | | КП.ІП-5225.045490.05.34 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 3 |

Form

Information source: ratesapi.io
Currency pair to analyze: EURUSD
Period to analyze: 2005-01-01 - 2006-01-01
SUMMARY
Annualized Return -0.133826
Annualized Sharpe Ratio -0.675730
Annualized Standard Deviation 0.116241

Analytical Results

Source ratesapi.io
Left Currency EUR
Right Currency USD
Start Date 01.01.05
End Date 01.01.06

Get information Back To Main Page

Рисунок 3 - Результати аналітичного та статистичного аналізу

При натисканні на кнопку “Charts” користувач переходить на сторінку візуалізації статистичних показників (рисунок 4).

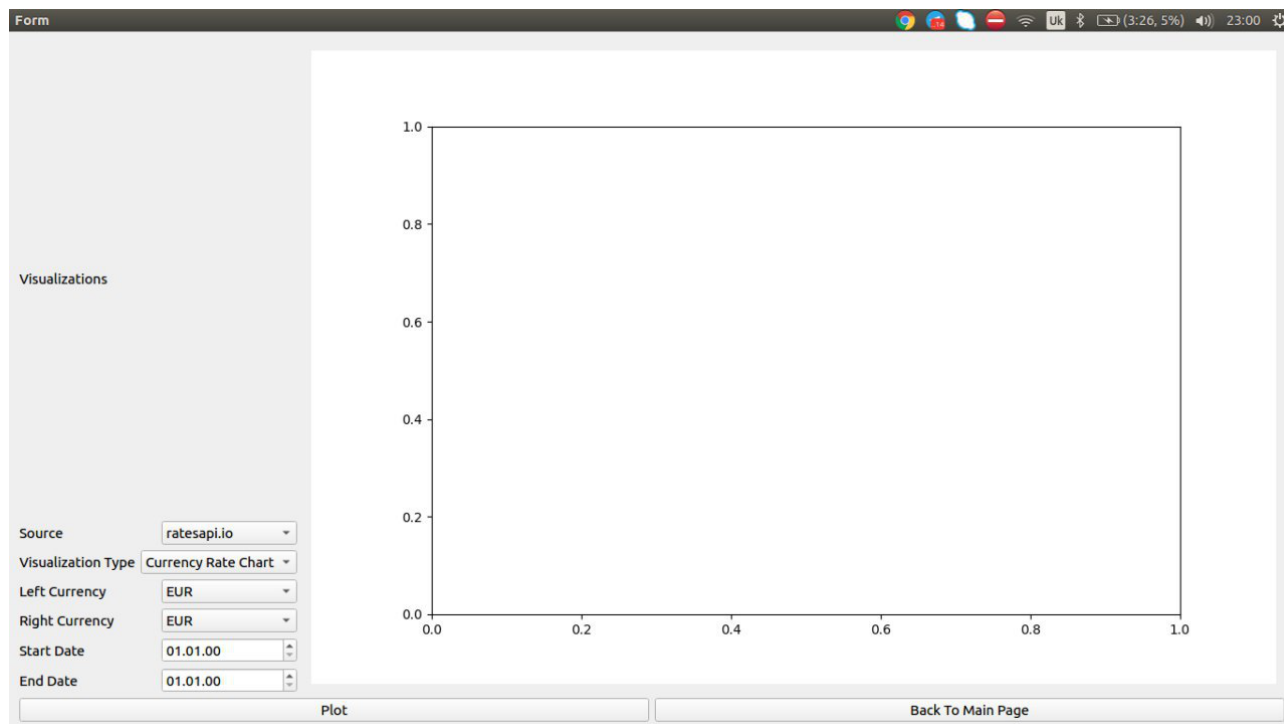


Рисунок 4 - Сторінка візуалізації статистичних показників

В лівому нижньому куті знаходяться поля вибору вхідних параметрів для аналізу. Необхідно обрати джерело даних в полі “Source”, тип візуалізації в полі “Visualization Type”, ліву і праву ногу валютної пари в полях “Left Currency” і “Right Currency” відповідно, початок і кінець періоду часу для аналізу в полях “Start Date” і “End Date” відповідно.

Для отримання результату необхідно натиснути кнопку “Plot”.

Якщо обраний некоректний період часу для аналізу (значення поля “Start Date” більше або рівне значенню поля “End Date”) буде виведене відповідне повідомлення.

Якщо ліва і права нога валютної пари однакові в поле результату буде виведене відповідне повідомлення.

Якщо обрані параметри коректні, то в полі візуалізації відобразиться обраний тип зображення, що враховує обрані обмеження. Приклади зображення візуалізацій для типів “Currency Rate Chart”, “ACF Plot”, “Snail Trail Plot”, “ECDF Plot”, “QQ plot”, “Drawdown plot” наведені на рисунках 5-10 відповідно.

Для повернення на головну сторінку сервісу (рисунок 1) необхідно натиснути на кнопку “Back to Main Page”.

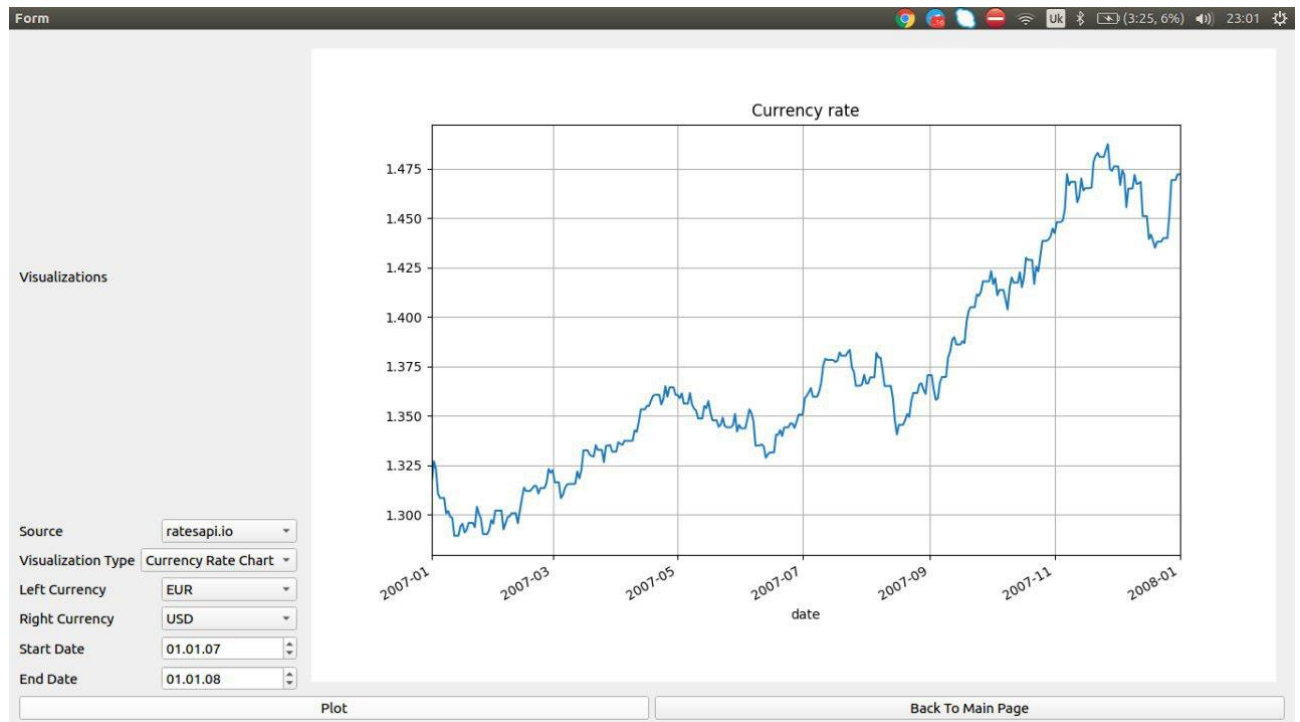


Рисунок 5 - Приклад візуалізації типу “Currency Rate Chart”

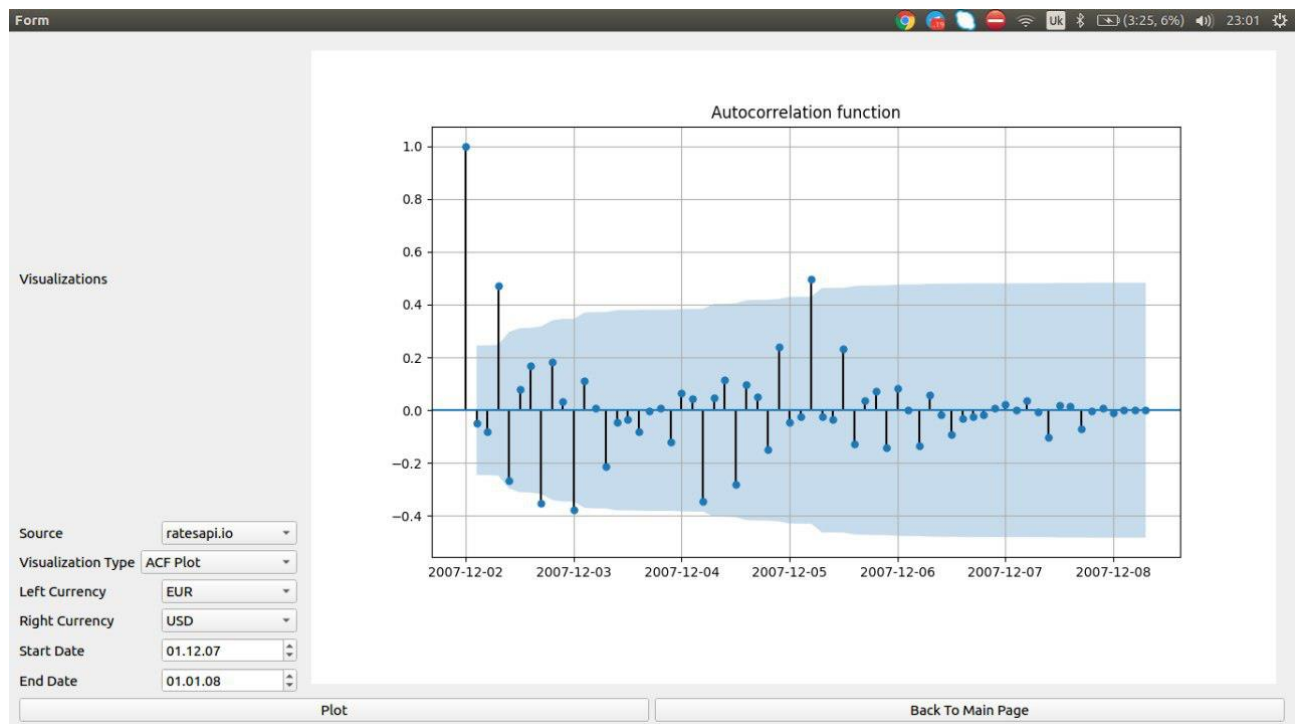


Рисунок 6 - Приклад візуалізації типу “ACF Plot”

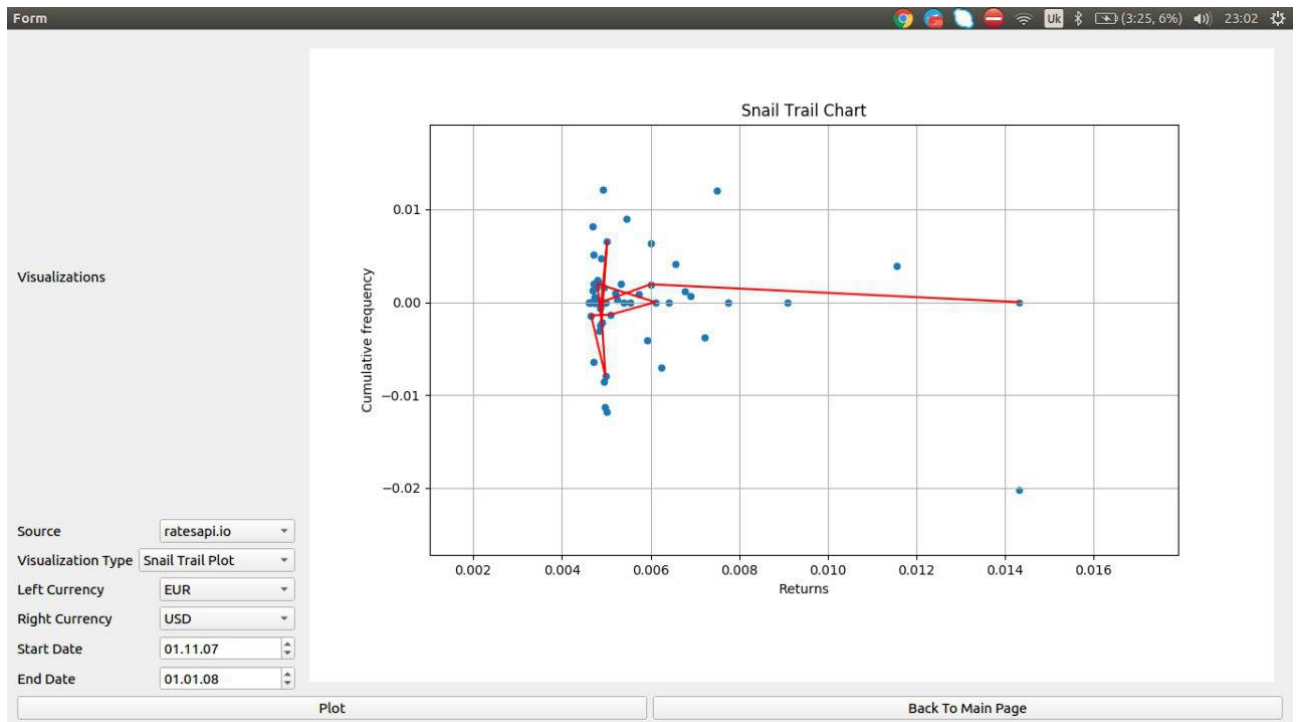


Рисунок 7 - Приклад візуалізації типу “Snail Trail Plot”

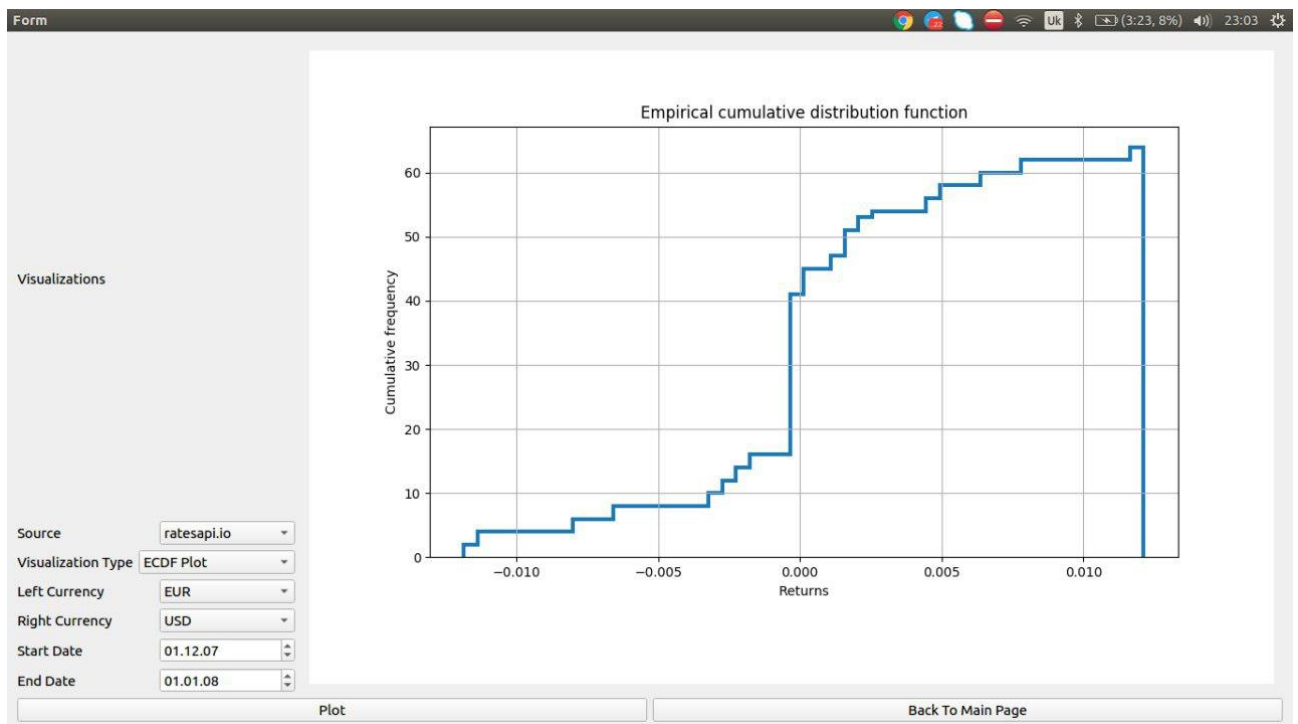


Рисунок 8 - Приклад візуалізації типу “ECDF Plot”

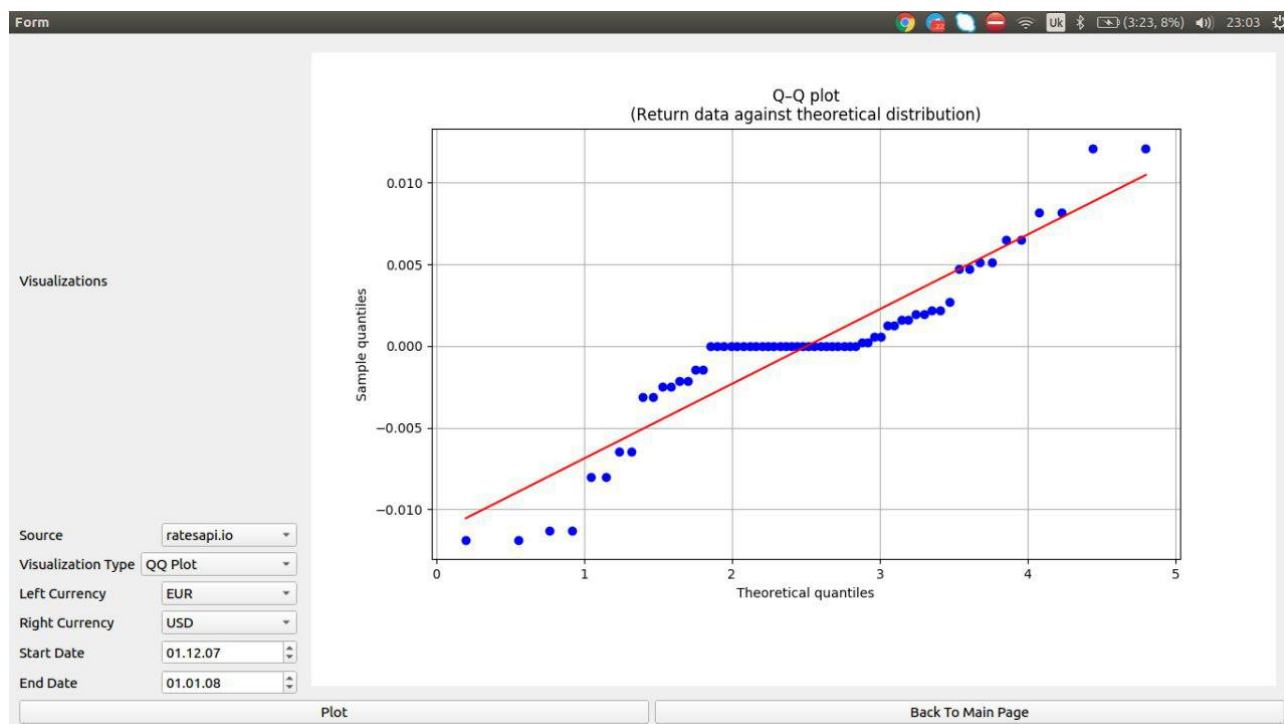


Рисунок 9 - Приклад візуалізації типу “QQ plot”

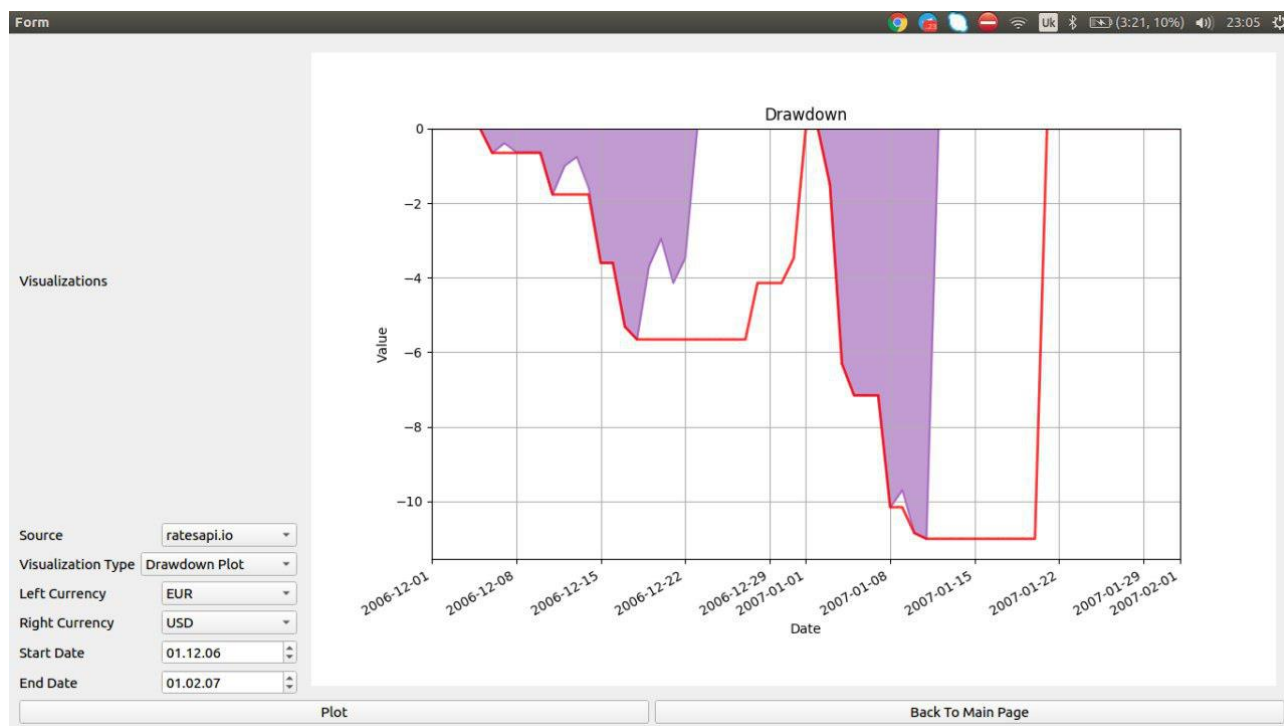


Рисунок 10 - Приклад візуалізації типу “Drawdown plot”

Для повернення на головну сторінку сервісу (рисунок 1) необхідно натиснути на кнопку “Back to Main Page”.

При натисканні на кнопку “Load data” користувач переходить на сторінку експорту курсу валюти за вказаний період (рисунок 11).

Рисунок 11 - Сторінка експорту курсу валюти за вказаний період

Зверху знаходиться кнопка вибору директорії для завантаження “Choose directory to load”. При натисканні на неї відкривається системне вікно навігації між директоріями (рисунок 12). Необхідно обрати шлях для збереження файлу і натиснути кнопку “Open” в правому нижньому куті.

Нижче знаходяться поля вибору вхідних параметрів для аналізу. Необхідно обрати джерело даних в полі “Source”, ліву і праву ногу валютної пари в полях “Left Currency” і “Right Currency” відповідно, початок і кінець періоду часу для аналізу в полях “Start Date” і “End Date” відповідно, формат файлу для збереження даних в полі “File Format”.

Для завантаження файлу необхідно натиснути кнопку “Load information”.

Якщо обраний некоректний період часу для аналізу (значення поля “Start Date” більше або рівне значенню поля “End Date”) в поле результату буде виведене відповідне повідомлення.

Якщо ліва і права нога валютної пари однакові в поле результату буде виведене відповідне повідомлення.

Якщо обрані параметри коректні, то файл з інформацією буде збережено в обраній директорії зі вказаним форматом (рисунок 13).

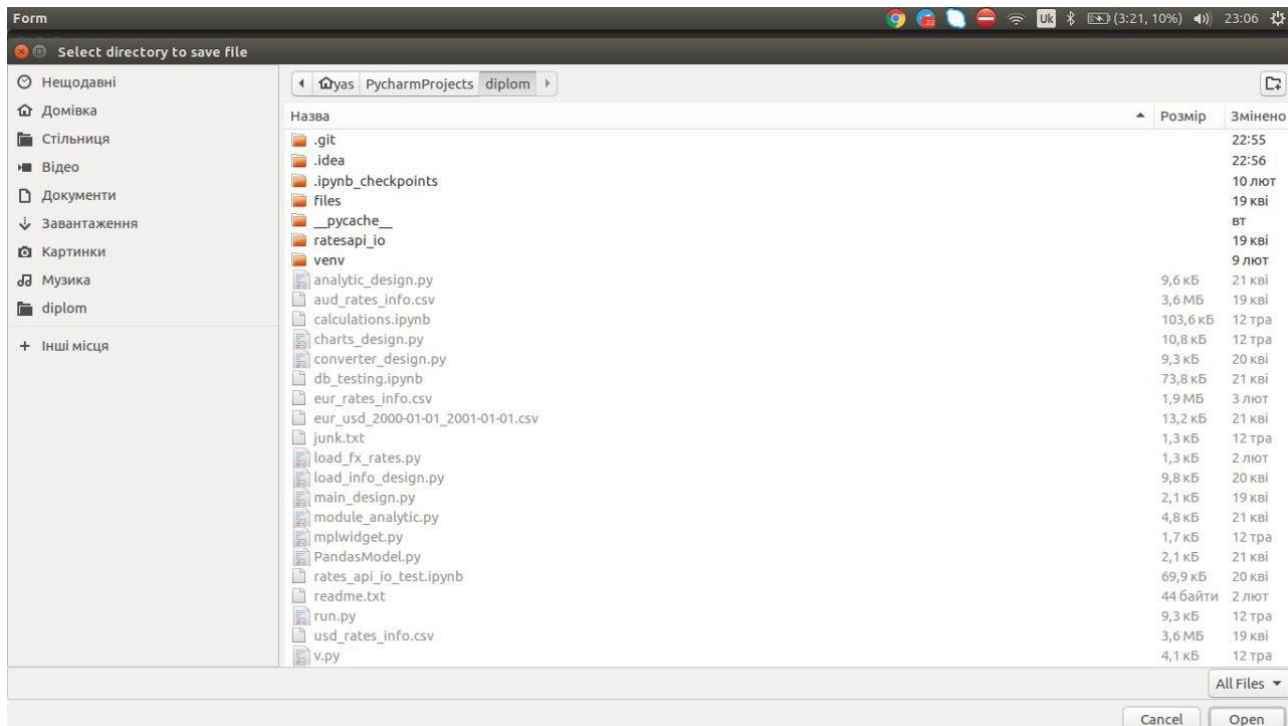


Рисунок 12 - Системне вікно навігації між директоріями

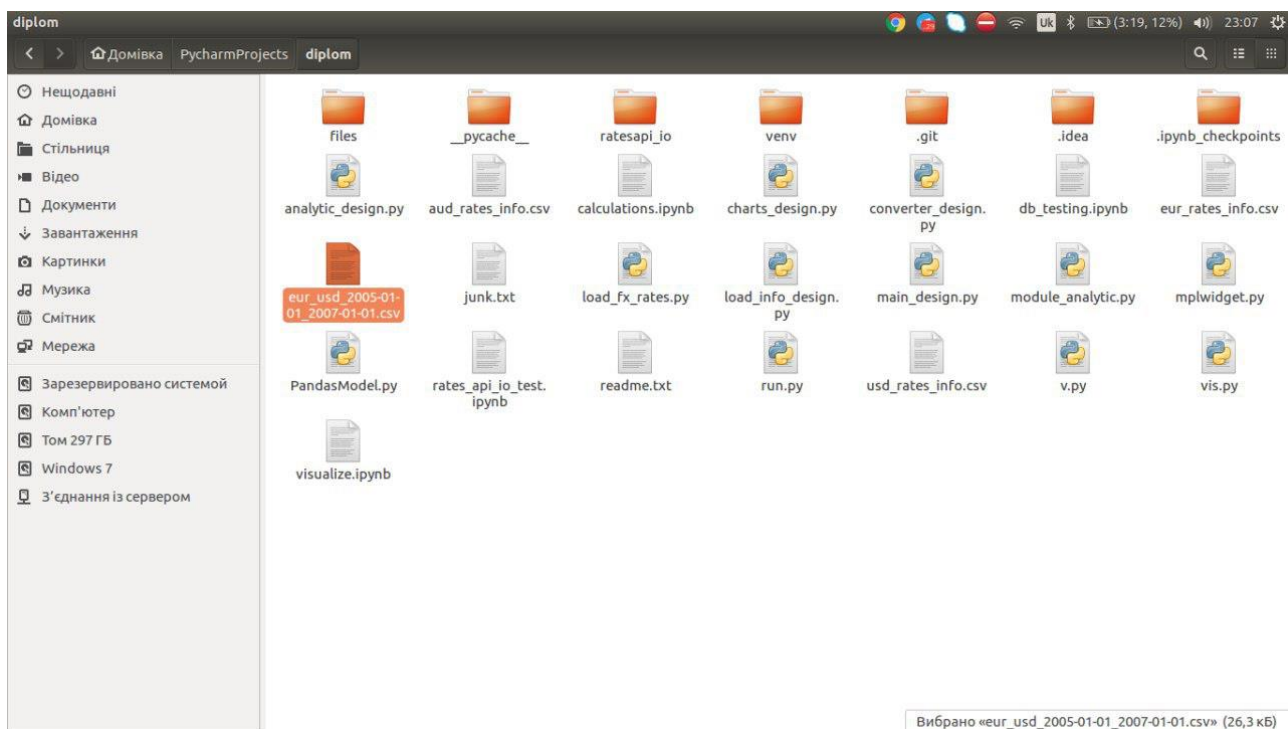


Рисунок 13 - Файл з інформацією збережено

| | | | | | | |
|------|------|----------|--------|------|--------------------------|------|
| | | | | | КПІ.ІП-5225.045490.05.34 | Арк. |
| Змн. | Арк. | № докум. | Підпис | Дата | | 10 |

Для повернення на головну сторінку сервісу (рисунок 1) необхідно натиснути на кнопку “Back to Main Page”.

При натисканні на кнопку “Currency converter” користувач переходить на сторінку експорту курсу валюти за вказаний період (рисунок 14).

Рисунок 14 - Сторінку експорту курсу валюти за вказаний період

На сторінці необхідно обрати джерело даних в полі “Source”, ліву і праву ногу валютної пари в полях “Left Currency” і “Right Currency” відповідно, і кількість валюти для конвертації в полі “Amount”.

Для перегляду результатів розрахунку необхідно натиснути кнопку “Convert”.

Якщо ліва і права нога валютної пари однакові в поле результату буде виведене відповідне повідомлення.

Якщо немає доступу до стороннього джерела поточного курсу валют також буде виведене відповідне повідомлення.

Якщо обрані параметри коректні і з'єднання з джерелом даних встановлене, то в текстовому полі відобразяться вхідні параметри та результати розрахунку конвертації між валютами (рисунок 15).

Currency converter

Source: ratesapi.io

Convert From: EUR

Convert To: RUB

Amount: 5,00

Buttons: Convert, Back To Main Page

Results:

```

Currency converter.
5.0 EUR is equal to 5.594 USD.
Currency converter.
5.0 EUR is equal to 8.103 AUD.
Currency converter.
5.0 EUR is equal to 360.676 RUB.
  
```

Рисунок 15 - Результати розрахунку конвертації між валютами

Для повернення на головну сторінку сервісу (рисунок 1) необхідно натиснути на кнопку “Back to Main Page”.

Для завершення роботи з додатком потрібно натиснути лівою кнопкою миші на хрестик у верхньому лівому кутку екрана.

ФАКУЛЬТЕТ ІНФОРМАТИКИ ТА ОБЧИСЛЮВАЛЬНОЇ ТЕХНІКИ
Кафедра автоматизованих систем обробки інформації і управління

«ЗАТВЕРДЖЕНО»

В.о. завідувача кафедри

_____ І.П.Муха
(підпис) (ініціали, прізвище)

“ ” _____ 2019 р.

«Сервіс аналізу ринку іноземних валют»

Графічний матеріал

КП.ІП-5225.045490.06.99

“ПОГОДЖЕНО”

Керівник проекту:

_____ Халус О.А.

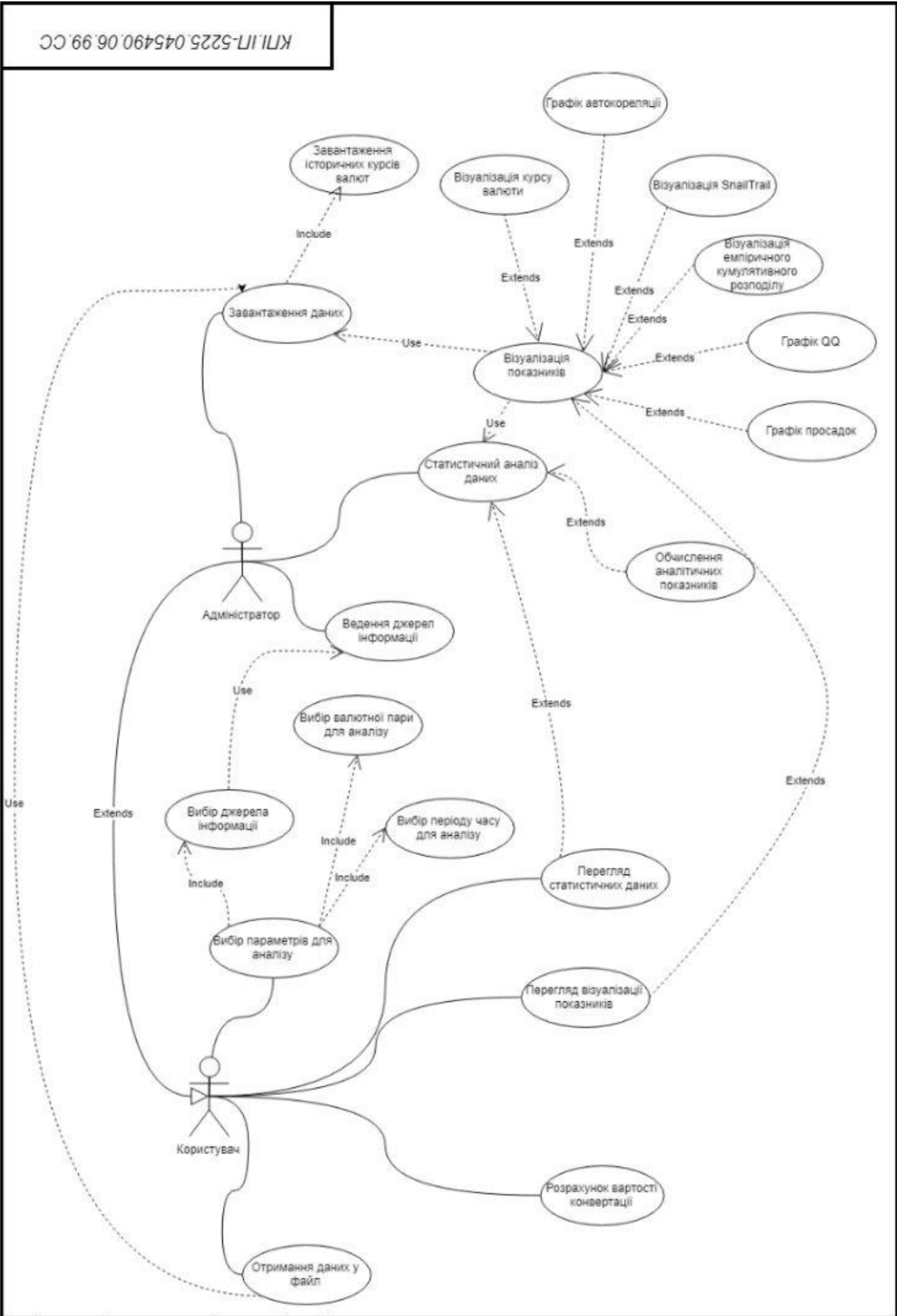
Нормоконтроль:

_____ Ліщук К.І.

Виконавці:

_____ Ясенова А.В.

Київ – 2019 року



| | | | | | | | | | | |
|---------|--------------|----------|-------|------|---|--|--|----------|--------|--|
| | | | | | КПІ.ІП-5225.045490.06.99.СС | | | | | |
| | | | | | Схема структурна варіантів використань | Лит. | | Арк. | Аркуші | |
| Зм. | Арк. | № докум. | Підп. | Дата | | | | | 1 | |
| Розроб. | Ясенова А.В. | | | | | | | | | |
| Перев. | Халус О.А. | | | | | | | | | |
| Т. Кон. | | | | | | | | | | |
| | | | | | | Аркуш 1 | | Аркуші 1 | | |
| Н. Кон. | Піщук К.І. | | | | Сервіс аналізу ринку іноземних валют | КПІ ім.Ігоря Сікорського Кафедра АСОІУ гр. ІП-52 | | | | |
| Затв. | Халус О.А. | | | | | | | | | |

Analytic

Charts

Load data

Currency Converter

Form

Visualizations

Source

ratesapi.io

Visualization Type

ACF Plot

Left Currency

EUR

Right Currency

USD

Start Date

01.12.07


End Date

01.01.08

Plot

Autocorrelation function

Back To Main Page



Form

Currency converter

Source

ratesapi.io

Convert From

EUR

Convert To

EUR

Amount

0,00

Convert

Back To Main Page

Form

Load info

Choose directory to load

Source

ratesapi.io

Left Currency

EUR

Right Currency

EUR

Start Date

01.01.00

End Date

01.01.00

File Format

xlsx

Load Information

Back To Main Page

| | | | | | | | |
|-----------|------|--------------|--------|------|--------------------------------------|--|-----------|
| | | | | | КПІ.ІП-5225.045490.06.99.KE | | |
| | | | | | | | |
| | | | | | | | |
| | | | | | | | |
| Зм. | Арк. | № документа | Підпис | Дата | Креслення вигляду екранних форм | Літера | Маса |
| Розробив | | Ясенова А.В. | | | | | Масштаб |
| Теревірів | | Халус О.А. | | | | | |
| Т. кон. | | | | | | | |
| | | | | | Сервіс аналізу ринку іноземних валют | Аркуш 1 | Аркушів 1 |
| Н. кон. | | Ліщук К.І. | | | | КПІ ім.Ігоря Сікорського Кафедра АСОІУ гр. ІП-52 | |
| Затвердив | | Халус О.А. | | | | | |